

DOCUMENT RESUME

ED 292 985

CE 049 918

AUTHOR Leonard, Janet
TITLE Wordprocessing and Language Skills. A Practical Handbook for ABE Tutors.
INSTITUTION Adult Literacy and Basic Skills Unit, London (England).
REPORT NO ISBN-0-906509-87-4
PUB DATE Jun 87
NOTE 134p.
AVAILABLE FROM Adult Literacy & Basic Skills Unit; Kingsbourne House, 229/231 High Holborn, London WC1V7DA, England (3 pounds).
PUB TYPE Guides - Classroom Use - Guides (For Teachers) (052)
EDRS PRICE MF01/PC06 Plus Postage.
DESCRIPTORS *Adult Basic Education; Adult Literacy; Bilingual Education; Computer Uses in Education; Foreign Countries; Illiteracy; *Language Skills; Learning Activities; Reading Comprehension; Reading Skills; Teaching Guides; *Word Processing; *Writing Skills

ABSTRACT

This practical manual is designed for teachers involved in adult basic education (in particular, teachers of bilingual and adult literacy students) who would like to introduce their students to word processing through the language curriculum. A range of activities linking language and word processing skills is suggested. The tasks can be approached either as stimuli for learning word processing skills or as language exercises that happen to make use of a word processor. All the activities are accompanied by suggestions as to how they can be integrated into the language curriculum. For each of the 30 tasks, the following components are provided: overview of the tasks, word processor task, language task, pre-computer work for students and teacher, follow-up exercise(s), materials the teacher should prepare, and notes to the teacher. Representative language tasks are simple creative writing, paragraphing, punctuation, upper/lower case, parts of speech, multiple choice--structure or vocabulary, proofreading, refining text, cloze, the language of questions and answers, vocabulary extension, writing a curriculum vitae, letter writing, sequencing, reading comprehension, homophones, roots and suffixes, summary writing, group composition, style and register, and spelling. Appendixes include advice on the choice of a suitable word processor and printer and a glossary of terms. (YLB)

* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *

ED 292 985

516 640 35

WORD PROCESSING AND LANGUAGE SKILLS

a practical handbook for ABE tutors

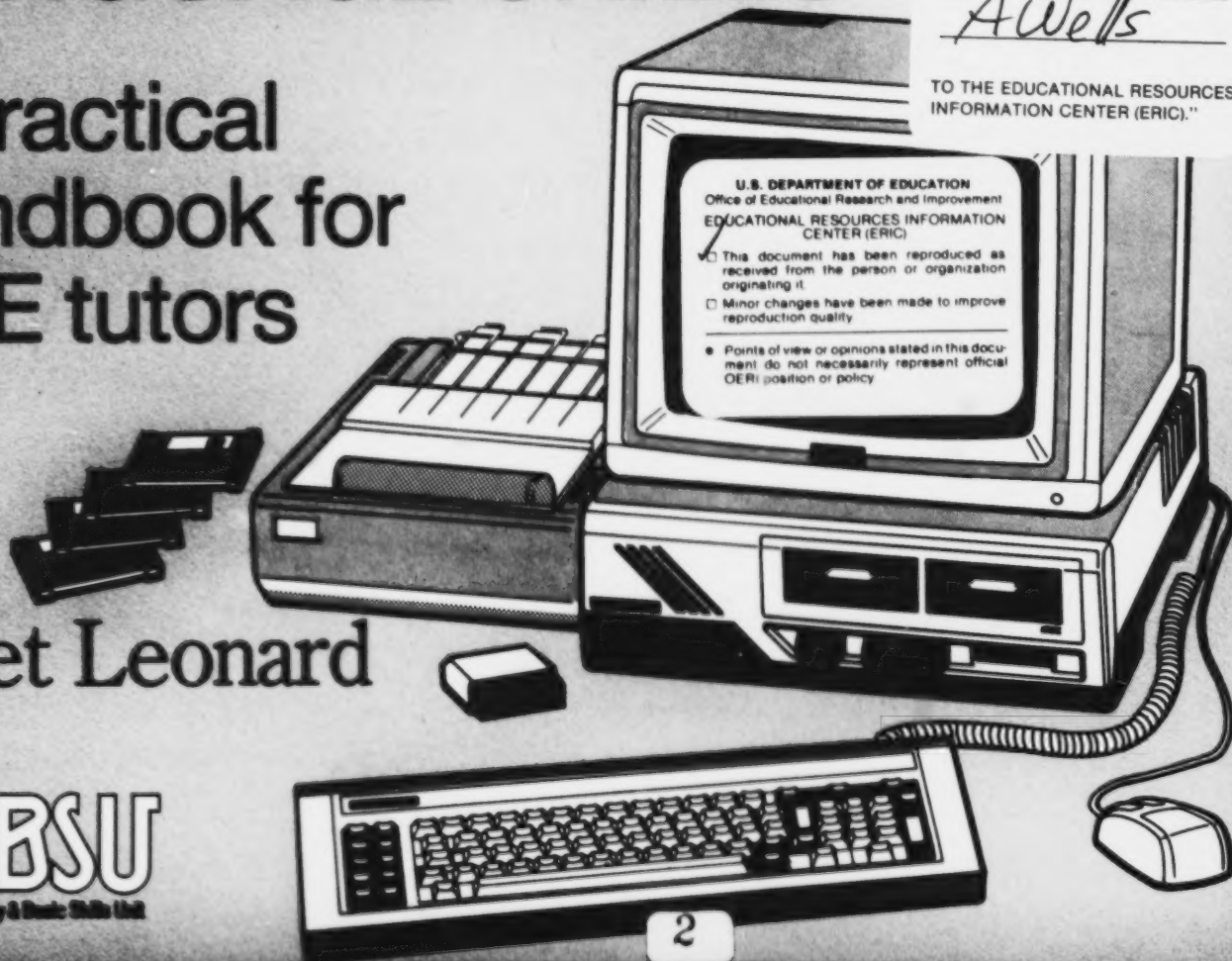
by Janet Leonard

ALBSU
Adult Literacy & Basic Skills Unit

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

A Wells

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."



PREFACE

Wordprocessing and Language Skills is a practical manual for teachers involved in adult basic education (in particular, teachers of bilingual and adult literacy students) who would like to introduce their students to wordprocessing through the language curriculum.

The book suggests a range of activities linking language and wordprocessing skills, and may be used both by teachers who are beginners to wordprocessing and by those who would like to apply their own knowledge of wordprocessing in the classroom.

The tasks can be approached either as stimuli for learning wordprocessing skills, or as language exercises which happen to make use of a wordprocessor. All the activities are accompanied by suggestions as to how they can be integrated into the language curriculum, and advice is given on the choice of a suitable wordprocessor and printer.

Janet Leonard is Project Co-ordinator of an ALBSU Special Development Project, established jointly with the Inner London Education Authority to provide in-service training and curriculum development opportunities in the field of computing for adult bilingual and literacy students. She is the author of *Computers in Language and Literacy Work*, published jointly by ALBSU and ILEA.

WORDPROCESSING AND LANGUAGE SKILLS

A Practical Handbook for ABE Tutors

by Janet Leonard

ACKNOWLEDGEMENTS

Thanks are due to all those who offered suggestions and advice on the text, especially to Mike Williams (Westminster College), John Taylor (Blackburn College) and Stephen Davies (ITEM-ILECC) for their constructive criticism, and to tutors on the ALBSU/ILEA Computer Development Project, some of whose ideas are incorporated in the Tasks.

© The Adult Literacy and Basic Skills Unit,
Kingsbourne House, 229/231 High Holborn, London WC1V 7DA

All rights reserved. No part of this publication may be photocopied, recorded or otherwise reproduced, stored in a retrieval system or transmitted in any form or by any electronic or mechanical means without the prior permission of the copyright owner.

Published June 1987

ISBN 0 906509 87 4

Design: Studio 21

CONTENTS

Introduction

Why use a wordprocessor?	7
Wordprocessing for students	8

Using this book

Worksheets and task-sheets	12
Approaches to wordprocessing	13
(1) <i>The systematic approach</i>	14
(2) <i>Building on basic editing</i>	15
(3) <i>Random access</i>	15
(4) <i>Multi-media</i>	15
(5) <i>Self-access</i>	16
Wordprocessing across the curriculum	17
Equipment	17
A health and safety note	17

Task 1:

Word-wrap	
Use of the RETURN key	
<i>Simple creative writing in paragraphs</i>	18

Task 2:

Use of the RETURN, DELETE and cursor keys	
<i>Paragraphing</i>	20

Task 3:

Inserting characters	
Naming, saving and printing files	
<i>Punctuation</i>	23

Task 4:

Case swap	
Loading files	
<i>Use of upper/lower case</i>	27

Task 5:

Whole-word deletion	
<i>Parts of speech</i>	30

Task 6:

Whole-word deletion	
<i>Multiple-choice – structure and/or vocabulary</i>	32

Task 7:

Consolidation – cursor movement and deletion/insertion	
<i>Proofreading for elementary errors</i>	35

Task 8:

Consolidation – cursor movement and deletion/insertion	
<i>Proofreading from a fair copy</i>	37

Task 9:

Copy-typing	
Consolidation – cursor movement and deletion/insertion	
Copying	
Proofreading	38

Task 10:

Consolidation – cursor movement and deletion/insertion	
(Deletion of phrases)	
Refining text – tautology	40

Task 11:

Overtyping/insert	
Cloze	42

Task 12:

Overtyping	
Form-filling	
"Audiotyping"	
The language of questions and answers	45

Task 13:

Using tabs	
Typing a table	
Vocabulary extension	49

Task 14:

Using tabs	
Writing a C. V.	53

Task 15:

Tabs	
Paragraph indents	
Justification/Ragged Right	
Composing and typing a formal letter	57

Task 16:

Tabs	
Hanging indents	
Paragraphing	60

Task 17:

Tabs	
Hanging indents	
Writing a questionnaire	63

Task 18:

Centring	
Underlining	
Italics, etc	
Invitations – cards and letters	
Letters of reply to invitations	66

Task 19:

Block move	
Sequencing	70

Task 20:

Block move	
Paraphrasing/register	73

Task 21:

Search and replace	
Information gap	
Reading comprehension	76

Task 22:

Search and replace	
Homophones	81

Task 23:

Search and replace	
Vocabulary building/idioms	83

Task 24:

Search	
Roots and suffixes	87

Task 25:

Block copy	
Windows	
Summary writing/précis	90

Task 26:

Merging files	
Group composition	96

Task 27:

Merging files	
Glossary	
Block move	
Rebuilding a text	98

Task 28:

Merging files	
File copying	
Writing up a discussion	101

Task 29:

Mailmerge	
Style and register	103

Task 30:

Spellcheck/dictionary	
Spelling	112

Appendix A

Choosing a wordprocessor	115
--------------------------------	-----

Appendix B

Choosing a printer	123
--------------------------	-----

Appendix C

Glossary of terms	127
-------------------------	-----

INTRODUCTION

Any microcomputer can become a wordprocessor, simply by loading a wordprocessing program into it. A wordprocessor is an "intelligent typewriter", which allows the writer to correct and manipulate text on a screen before printing it on paper.

The facilities offered by a wordprocessor depend as much on which wordprocessing program you use as on which computer you have. Whatever the computer, it is likely that there are several wordprocessing programs to suit it, so it is a good idea to try out a number of different ones before committing yourself irrevocably, especially if your choice is likely to be expensive¹.

Why use a wordprocessor?

Most people who have learned to use a wordprocessor would not willingly return to using an ordinary typewriter.

To any typist, the wordprocessor gives the freedom of altering and re-working a piece of writing before committing it to paper.

For the two-finger typist, there is the opportunity to correct typing mistakes with ease, without having to resort to correction-fluid, or having to re-type the whole piece.

Everybody at some time needs to re-format a page of text that is just one line too long to fit on the page: with a wordprocessor, the margins can be reset, a blank line removed, or a paragraph reworded, just to make enough space.

For the teacher preparing worksheets, the wordprocessor has innumerable advantages over a typewriter, most of which help to make life easier and cut down on typing time. For example:

1. Appendix A discusses some of the features of wordprocessing programs, and offers advice on points to look for when making a choice. Appendix B deals with the choice of a printer.

- each piece of typing can be saved, so that it can be reprinted as many times as necessary in the future, without any further typing; texts can be printed directly on to a banda-master, so that you can have a fresh class-set any time you need one;
- worksheets can be edited, so that a question which proved too difficult can be deleted or reworded, or an unnoticed error which slipped through on the first typing can be removed for future printouts;
- perhaps your text mentions the name of an individual (a student's name, or the name of a public figure, for example) which is no longer relevant. Just swap it for another name: a single command on the wordprocessor will allow you to change every occurrence of the name throughout the text;
- perhaps you want a blank question-sheet, and another version with model answers typed in: just type your question-sheet, and copy it on to a new page – almost instantaneously you have two identical pages, and you may then type your answers on to the new page, using the wordprocessor's "overtyping" facility.

The wordprocessor can help to make the tutor's life easier by offering a range of text-handling facilities which the humble typewriter could never hope to equal.

What does it offer the student?

Wordprocessing for students Seymour Papert¹ writes:

For me, writing means making a rough draft and refining it over a considerable period of time. My image of myself as a writer includes the expectation of an "unacceptable" first draft that will develop with editing into presentable form ... (but) for most children rewriting a text is so laborious that the first draft is the final copy, and the skill of rereading with a critical eye is never acquired.

Most adults are also in this position. Re-drafting is a skill which is never acquired, because it is tedious and time-consuming. For students writing in a language which is not their native one, it may

1. Papert, Seymour **Mindstorms: Children, Computers and Powerful Ideas**. The Harvester Press, Brighton, 1982

be the physical difficulty of coping with the handwritten version of a strange alphabet which is a problem, as much as the lack of language skill. This may also be true, of course, for adult literacy students who are not accustomed to writing. The ease of editing on a wordprocessor can be an incentive to read, re-read, consult others, and make improvements on an original text, or perhaps try out several different versions.

There is also the ability to save a text to return to at a later stage, without the pressure of feeling that it has to be finished at one sitting. This also gives an opportunity for thought away from the computer, which may lead to new ideas and a subsequent revision which would never be undertaken if it involved starting again from scratch.

Many students believe that mistakes are "bad", and should not be made. This can be very inhibiting. The wordprocessor is a great leveller in this respect: it can demonstrate better than any other medium that "mistakes" are part of the creative process, and that everybody makes them, whether they are beginner-writers, teachers, or Shakespeare. Indeed, "a person who has not made a mistake, has made nothing". Having discovered that, it is nevertheless comforting to know that our mistakes on a wordprocessor can be eradicated so that subsequent readers need never know we made them.

Wordprocessing and handwriting

Some teachers (of infants, notably) have expressed fears that use of the wordprocessor will adversely affect handwriting ability. There is no evidence that this is so, and in fact the evidence points the other way – that wordprocessing can actually stimulate efforts with a pen. This is probably more so for adult literacy students than for children. It is important to distinguish here between the skill of handwriting, and the skill of composing. Many adults are inhibited in their composition skills because they have difficulty with handwriting and with spelling. This leads to the frustration of ideas outstripping the speed of the pen, and the belief that they "can't write" is applied equally to the physical and the mental processes.

Even though typing is a skill in itself, it is easier to acquire than the skill of good, fast handwriting. And with the ability of even a hesitant typist to correct at will, comes the ease of producing a good clean text, written in neat lines without crossings-out. Many adults coming to wordprocessing for the first time suddenly discover that, freed from the need to concentrate on what the pen is doing, they have plenty of ideas about what to write; and freed from the necessity to get the spelling right (they can always come back and correct it later), they can write far more than they ever believed possible. There is then an incentive to improve the

handwriting, because students can appreciate the distinction between having something to write, and being able to write it.

There is therefore a justification for using a wordprocessor with both bilingual and literacy students, if only because it simplifies the writing process, and allows freedom of composition and correction which the pen alone can not offer.

The great thing about wordprocessing is its versatility. It can help a literacy student to learn skills of composition, and even of spelling. It can help a bilingual student to practise the structure and behaviour of the English language. It can help students in any area of the curriculum to learn a skill (wordprocessing), while at the same time learning about or practising whatever subject they are most interested in.

A wordprocessor is a tool, and learning how to use it is not an end in itself: we do not learn how to use a pencil for its own sake, and we should not approach wordprocessing in that way, either. Students can use wordprocessing as a tool to help them write more, and to edit what they write so that the printed version is something they can feel satisfied with.

If you are still tempted to ask "Why not use a pencil and paper?", here are some more good reasons why a wordprocessor can often be a better choice:

- there is a motivation about using a wordprocessor that is impossible to equal with traditional writing tools; the wordprocessor encourages perfectionism – an inclination to get things right;
- the ease of obtaining printed copies of wordprocessor texts encourages students to proofread, revise, then return to the computer for another go – and a better printout; it also makes it easier to share written work, and the possibility of an audience can be a motivating factor;
- the vertical format of the screen makes group- or pair-work in writing tasks much easier to achieve than when using pencil and paper;
- learning a new skill induces a feeling of self-respect.

A further reason, if any is needed, is that in this latter part of the twentieth century, computer skills are – rightly or wrongly – very highly valued in our society. This is not to say that a person

who has learned wordprocessing can walk straight into a job, but it does mean at the very least an extra experience to offer to an employer, and an increase in confidence for the person who has learned a new skill.

Not all students will want to use a wordprocessor, and not all those who use one will want to repeat the experience. But it is nevertheless true that the majority of students who try wordprocessing benefit from it, and want to continue, and many of those who are hesitant at the outset become ardent converts before long. And the same is true of teachers, of course.

USING THIS BOOK

Before embarking on the tasks, the students should be familiar with the concept of wordprocessing, but need not know how to operate a computer or wordprocessor.

The exercises are designed to introduce students to a range of the functions which a wordprocessor can offer, through the medium of texts and language activities which will interest them and develop their language awareness. The early exercises are the most important ones in getting students accustomed to using the basic functions of the wordprocessor; the later ones tend to be more creative, and to give more opportunity for self-expression in a controlled environment. The tasks are not designed specifically for the purpose of teaching wordprocessing: if learning wordprocessing is the prime aim, the student would be better off attending a wordprocessing course. All the activities are directed primarily towards language acquisition.

A considerable amount of advance preparation will be required on the part of the tutor, but it is well worth the effort, because in this way students will be introduced to wordprocessing through their own curriculum, and will be learning about the facilities a wordprocessor can offer almost as a by-product of studying the behaviour of language. Many of the language exercises suggested are ones which the students will probably need to do in some form or other in any case. The time taken in preparing materials initially will be repaid in the long-term because both exercises and handouts can be stored on disc and used over and over again, and can be easily modified to suit a particular student or group of students.

Worksheets and task-sheets

Each new task will involve two different kinds of information: information about what the task is (for example, writing a C.V., doing a cloze test, multiple choice), and information about the wordprocessor function required in order to carry out the task (for example, using tabs, overtyping, block move). These two kinds of information should be kept completely separate. Information about the task itself can be written up briefly in a task-sheet if you think it is necessary, or can simply be given orally before embarking on the activity. Worksheets should cover specific details of the

wordprocessor function involved in the task, and these can then be retained by the student for future reference. This means that, while worksheets must obviously contain enough information to enable students to carry out the specific task in hand, they should also be sufficiently generalised as to allow students to operate that particular wordprocessor function in another environment at a later stage.

The tasks are carefully graded so as to introduce new concepts of wordprocessing in easily-absorbed stages. Part of the value of the worksheets which you create to accompany each exercise is their use as reading comprehension material. They should therefore be written carefully to match the reading abilities of the target students (this may mean that you have to write more than one version to suit different groups of students). Students can store their worksheets in a ring-binder, which will eventually result in each student accumulating a personal wordprocessor manual written at an appropriate level of language.

The worksheets referred to in the sections marked **To prepare:** are only *new* worksheets, and the assumption is made that students will already have covered previous tasks and will therefore have access to earlier materials. If the exercises are not tackled in order, care should be taken to provide any earlier worksheets which are necessary to the activity, and careful checks should be made to ensure that students are not being asked to cope with too many new functions at once.

Approaches to wordprocessing

There are a number of ways of using the materials you prepare, and the method you choose for your students will depend on the type of students involved, their individual needs, and the time and equipment available. It is very difficult to give a timescale, as so much depends on the individuals, and how many of the supporting activities are undertaken, but the tasks suggested in this book would probably extend over a period of two or three terms, spending two or three hours a week on wordprocessing and related work. Perhaps only half an hour or so of this will be spent at the computer (more for the later tasks). It is possible, however, to achieve a moderate competence in wordprocessing, sufficient to start some creative writing, after only an hour or so at the keyboard.

Whichever of the following methods you decide on, practice is essential. For the students, this is less important, as many of the exercises they do on the computer can be considered as free-standing, but for the tutor who has to be ready to help out with problems as they arise, there is no substitute for frequent keyboard practice until the commands of the wordprocessor become second nature.

One method of getting into wordprocessing is to sit at the machine and type, learning a few basic editing techniques as you go along, and being content with what is, in effect, a typewriter with rather superior correction facilities. There is nothing wrong with this method, but it misses out on many of the useful and interesting features of a wordprocessor. And if students wish to learn wordprocessing with a view to improving their job opportunities, this is not the best method to use.

The following paragraphs describe some of the ways in which this book might be used in a more structured approach. Method (2) would be suitable for many students, especially those who are unable (or unwilling) to cope with the more advanced techniques, but would like to learn how to wordprocess for the purpose of creative writing. (4) and (5) will be of particular interest to those who have limited equipment and large classes. (3) stresses that the object is to practise language skills, wordprocessing being just another tool for this purpose. In fact, the language element comes first, no matter which method is used. The computer is merely a means to an end, and the end is language-acquisition.

1. The systematic approach The tasks in the book are graded according to the wordprocessor technique they introduce. This means that it would be possible to start at Task 1 and work through the book from start to finish, gradually learning the features of the wordprocessor in a systematic way, so that in the end students would have all the means at their disposal to use the wordprocessor as a creative writing tool.

Advantages

- students are introduced to most of the useful features of a wordprocessor, through their own curriculum;
- they learn the features stage by stage, each new task building on the previous ones, with plenty of reinforcement;

Disadvantages

- some of the language tasks may be inappropriate for some students, and the tutor would need to be selective;
- the language tasks may appear in an order in which you would not normally introduce them;
- the wordprocessor tasks, though graded as carefully as possible, may not be in a suitable order for the wordprocessing program you are using (for example, *paragraphing* is introduced in Task 1, but paragraphing on *Edward* is not as easy as on most wordprocessors, and may need to be held back to a later stage);

- some of the tasks may be inappropriate for your wordprocessor; the most obvious problems are pointed out in the text.

2. Building on basic editing

The opening tasks (say, numbers 1 to 7) could be used to give the students a grounding in basic editing sufficient for them to get started on creative writing; new techniques can then be added as the need arises.

Advantages

- the students quickly feel that they are achieving something creative, rather than just doing exercises;
- when they actually *need* to learn a new technique, they will be motivated to do the exercise associated with it;

Disadvantages

- some students may never get past the stage of basic editing, and will therefore miss out on the more interesting things that wordprocessors can do;
- if the later exercises are not tackled in order, care will have to be taken to ensure that students have covered all the earlier techniques which are necessary to whichever task they are doing.

3. Random access

A third possibility could be to approach wordprocessing from the language point of view, selecting tasks for their language content, rather than their wordprocessing content.

Advantage

- work on the wordprocessor can be integrated with the normal language curriculum;

Disadvantage

- students must have sufficient information about *all* the wordprocessor functions they will need in order to carry out each exercise (for example, Task 10 draws on most of the functions introduced in earlier tasks); this means very careful and detailed preparation of worksheets, and even so it may mean that some students will not be able to cope with the wordprocessing skills of the more sophisticated exercises, even if these are within their language capabilities.

4. Multi-media

It is quite possible to structure the task sheets in such a way that the exercises can be done either on the wordprocessor or with pencil and paper, and this means that in situations where equipment

is scarce, students can be doing the same exercise using different media, and different groups of students can use the wordprocessor from session to session, so that all those who want to use the machine eventually have a chance to do so.

Advantages

- with careful preparation of the task sheets, the work you prepare for the wordprocessor can be used by all the students in a group; this in itself can lead to discussion on different ways of achieving the same object, and on the pros and cons of using a wordprocessor;
- it is still possible to make use of the pre- and post-computer ideas;

Disadvantages

- it may well be, at least at first, that the students on the wordprocessor demand more attention than the other students;
- the other students usually want to know why they can't have wordprocessors, too.

5. Self-access

This method has been used successfully where only one computer was available in a supervised workshop setting. It means preparing in advance a wide range of exercises and a full set of worksheets for operating the wordprocessor. Students can then use whichever of the materials interest them, preferably guided by a teacher. It is essential to index all tasks twice, by language-content and by wordprocessor-content, and for each task to contain information directing students to all the worksheets necessary to carry out the activity.

Advantages

- students can work at their own pace, with their own choice of activity;
- they can choose whether to approach the work from the language or the wordprocessing point of view;

Disadvantages

- students working in this way need considerable teacher- support, which may tend to draw the tutor away from other students in the group, especially in the early stages;
- unless the student is prepared to accept guidance as to the order of the tasks, the work may be rather confused and aimless; in particular, Tasks 1 to 7 are essential before attempting later activities;
- the pre- and post-computer work, which is a valuable aspect of integrating the tasks into other classroom activities, will be lost.

**Wordprocessing
across the
curriculum**

The exercises in this book use elements of language-learning to demonstrate what can be done in the early stages of using a wordprocessor. The language-learning elements can, of course, be converted into similar exercises involving other areas of the curriculum. All the exercises should, in any case, be used as a springboard for further ideas on combining subject-study with the use of the wordprocessor.

Equipment

Many students feel more confident sharing a computer with another student; in any case, students working in pairs learn from each other, and the oral activity stimulated in this way can be valuable, especially for bilingual students. However, unless the activity is group composition, two at each machine should be the maximum for wordprocessing, so that each student can get as much experience of the keyboard as possible. As they become more confident and proficient, a machine for each student is, if not essential, at the very least highly desirable.

**A health and
safety note**

Working in front of a screen can be very tiring to the eyes, especially in the initial stages. Students should not sit too close to the screen (15" - 24" is a suitable distance), and should be encouraged to take quite frequent breaks whenever they feel the need to rest their eyes. The breaks can usually be turned to good account as discussion periods, but in fact most students are so keen to get on with the activity in hand that they do not like to spend too long away from the machine.

TASK 1

Wordprocessor tasks:
wordwrap
use of the RETURN key

Language task:
simple creative writing in
paragraphs

Before starting to edit texts, students will enjoy experimenting with some typing of their own, to experience word-wrap¹, and the use of the RETURN (or ENTER) key for paragraphing. Encourage the students to type as much as possible, and to ignore their mistakes (editing is introduced gradually in later tasks). One of the objects of the early exercises is to convince students that mistakes can be freely made and are not irretrievable. However, the students should not be encouraged to type anything of great importance to them in the early stages, as they will probably want to use more complex wordprocessor skills than they know how to achieve, and this can easily lead to frustration. A piece of the student's own writing which you have already discussed together might be a suitable starting point, with the student copying this on to the wordprocessor, and perhaps adding something extra in the process. A letter or C.V. would be too complicated at this stage, as these require a great deal of special formatting.

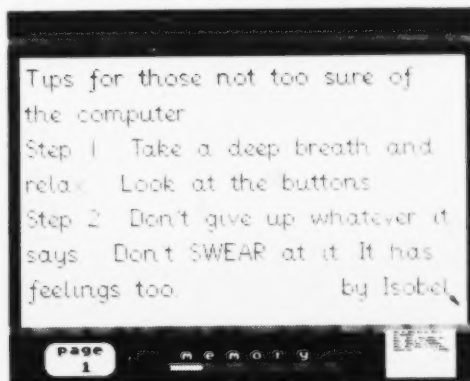


Fig. 1. Creative writing by a first-time user. *Folio* (BBC) in "display" mode.

¹ If you are not familiar with the term "word-wrap" – or with any other terminology you come across in the text – turn to the Glossary for an explanation.

The object of this first exercise is simply to get the student creating a personal piece of work on the wordprocessor. It will probably be slow, as most students will not have experience of a computer (or typewriter) keyboard, but for many students it will not be slower than handwriting. Later exercises approach the question of keyboard skills gradually, introducing different aspects of the keyboard in a controlled way, but for the purpose of this task, the student has the freedom to use whatever keys are necessary. You may want to introduce the use of the SHIFT key to make capital letters at this stage, or you could hold this over until its formal introduction in Tasks 2 and 3. Encourage the use of the RETURN key for paragraphing – at the end of every sentence, if the text is short.

When the student has finished writing, print out the text for follow-up discussion.

Pre-computer work:

- the students could prepare a piece of work specifically for this purpose – perhaps writing how they feel about using the wordprocessor for the first time;
- any piece of prepared writing that is going to be used on the wordprocessor should be discussed beforehand, and as many as possible of the language difficulties ironed out.

Follow-up exercise:

- discuss the printed text, encouraging the student to look for ways in which it might be improved, whether in layout, adding a title, use of upper and lower case, or typing errors;
- you may want to go back to the computer to demonstrate how the text could be edited (without entering into too much detail about how the editing is achieved at this stage);
- print out a final version for the student to take away.

To prepare:

- nothing, although you may wish to have the wordprocessor switched on and ready to start; even this might be part of the student's first lesson, if you have a wordprocessor which is easy to get "up and running" (although loading files is formally introduced in Task 4).

NOTES

Most people, however experienced they are at writing, find the facilities offered by a wordprocessor wonderful on meeting them for the first time. There is a temptation to want to show a newcomer everything the machine can do: resist the temptation! Too much in one go leads to mental indigestion.

TASK 2

Wordprocessor task:
use of the RETURN and DELETE
and cursor keys

Language task:
paragraphing

This exercise is a very simple introduction to editing, involving the use of only two keys, in addition to the cursor keys.

Prepare a wordprocessor file consisting of a text typed as a single unit (that is, without pressing the RETURN key), which is to be split into its constituent parts. The "constituent parts" can be anything from a single word to a paragraph in length.

At its most basic level, the text could be a list of simple instructions, or perhaps a shopping list, typed as a single paragraph to be divided into suitable units. Beginner readers might simply be asked to type their names (if possible, using upper and lower case appropriately), pressing the RETURN key after each name; the tutor could then delete all the RETURNS, making the names into a long "strip", which the students reconstitute into a list.

A further idea is to offer a poem with rhyming lines, presented as a piece of continuous text, the task being to split the poem into its constituent lines, using the rhymes as guides.

For more advanced students, the text can be fairly long with a definite storyline or argument, and the students' job will be to split it into paragraphs. The text should obviously be within the students' linguistic capabilities, but it should, if possible, be open to more than one interpretation of paragraphing. Students could discuss a printed handout of the text in groups, deciding where the paragraph breaks should be made before going to the wordprocessor to carry out the task. The discussion might involve arguing a case for alternative ways of paragraphing, using the linguistic and semantic clues in the text.

At this stage you should load the text into the wordprocessor yourself so that the students need only consider the exercise itself, not the technicalities of how to get it on the screen (this will be tackled at a later stage).

The only keys necessary for this exercise (whatever the level of difficulty) are

- the cursor keys (to position the cursor where the paragraph break should occur);
- the RETURN key (to make the break)¹;
- the DELETE key (to return to the status quo, should the break prove to be unsatisfactory).

Pre-computer work:

It is advisable, especially in the early stages of using the wordprocessor, to prepare the language task thoroughly before approaching the machine, in order to avoid confusing students by asking them to learn the wordprocessor and the language at the same time. Most of the types of text used for this exercise will therefore need to be previewed in some detail before the wordprocessing part of the activity begins:

- a shopping list might be based on the actual shopping needs of a particular student (or composite of a number of students);
- a list of instructions (for example, "How to make a pot of tea" or "How to start a car") could be discussed, agreed and then typed by the students themselves;
- the students may like to write their own poem as raw material for the rhyme exercise;
- the longer text for paragraphing could be selected from material which the students have already prepared, and pre-computer discussion could focus on how to go about the job of paragraphing in general, as well as the possible paragraph breaks of the chosen text.

Follow-up exercises:

- the use of a poem could spark off some intensive language work, based on the relationship between spelling and sound (using rhymes such as "bare", "there", "wear", "hair");
- if a narrative or argumentative text is chosen, one follow-up exercise might be to discuss the merits of the various interpretations of paragraphing, using printouts (made by the tutor – printing texts is introduced in the next task) of students' work.

1. Most wordprocessors use the RETURN key to split existing text into paragraphs, but one or two have specific keys for this purpose – for example, *Folio* uses a function key, and *Edword* uses a function key plus another key.

To prepare:

- a wordprocessor file consisting of a text typed as a single paragraph,
- a printed handout of the same text, if appropriate,
- a worksheet, covering the use of the cursor keys, and the RETURN and DELETE keys.

NOTES:

Although this task is not difficult in itself, the students will be involved in a number of processes which are unfamiliar to them. If they are bilingual students, there should be a considerable amount of oral work involved in carrying out these processes, in addition to the language work involved in the exercise itself. Do not underestimate the time it will take for students to complete the activity!

The exercise involves a number of basic wordprocessor techniques: the cursor keys, and the RETURN and DELETE keys are the most frequently used keys on the wordprocessor keyboard, and their use is an early priority.

One function of the DELETE key is difficult to grasp, and it would be useful to stress it in your worksheet. This is the use of DELETE to "undo" a RETURN. The use of DELETE to "undo" a letter of the alphabet, or any character which is visible on the screen, can easily be understood. As far as the wordprocessor is concerned, however, pressing the RETURN key is just the same as pressing any other key, and can therefore be undone just as readily with DELETE.

TASK 3

Wordprocessor tasks:
inserting characters
naming, saving and printing files

Language task:
punctuation

This task uses a text stripped of its punctuation. The exercise is to use the "insert" facility of the wordprocessor to put the correct punctuation in the text.

Prepare a text which is within the linguistic capabilities of the students, but omitting the punctuation. The text may be one which the students have already studied in another context, or another possibility is to use the students' own language-experience writing as source material for texts, although this will mean creating a separate text for each student doing the exercise. The language-task element can be varied in difficulty by grading the omission of punctuation – omitting only commas, for example, or removing full stops but leaving capital letters, or ultimately taking away all punctuation.

In addition to the RETURN, DELETE and cursor keys used in the previous task, students will need to be able to identify and use the punctuation keys on the keyboard. This may affect the level of the language task you are able to offer, so you should look carefully at your own keyboard before deciding which punctuation to strip from the text. Normally, the comma and full-stop characters are obtainable directly from the keyboard, without using the SHIFT key; some or all of the other punctuation marks will need the SHIFT key to be held down while the punctuation key is pressed. You may prefer at first to offer an exercise which involves only the punctuation characters which are obtainable without SHIFT. Whatever your choice, students should be conversant with the position on the keyboard of the punctuation marks they are going to use, and with the use of the SHIFT key if necessary.

The task involves positioning the cursor at the point where the punctuation is needed, and inserting the appropriate punctuation mark. Mistakes can of course be "rubbed out" with the DELETE key.

There are two modes of text-entry on a wordprocessor, and you will need to find out which of these

is the "default mode"¹ on your wordprocessor before preparing a worksheet for this task. Suppose you type the sentence:

There is a in this sentence

and then realise you have omitted the word "mistake". If you are using *Wordwise*, you will be able to move your cursor to the "i" of the word "in", and type "mistake", so that you will see on the screen:

There is a mistake in this sentence

This means that "insert mode" is the default in *Wordwise* – when you type something into the middle of existing text, everything that follows moves over to make room for the insertion.

With the BBC's *View* wordprocessor, on the other hand, if you use the same model sentence:

There is a in this sentence

and type the word "mistake" into it, the result will be:

There is a mistake sentence

– that is, the word "mistake" has wiped out the original words "in this", because in *View* "overtyping" is the default mode.

If the wordprocessor you are using has "overtyping" as its default mode, you will need to include in your worksheet for this exercise the command necessary to switch to "insert mode", so that the students can put punctuation into the text. You should then include at the end of the worksheet the command to switch back again to "overtyping mode".

When the punctuation task is finished, students should save and print their finished texts. Since punctuation can always bear a little extra practice, you could perhaps prepare several graded examples of texts to be punctuated (possibly covering more than one session on the computer), and introduce saving and printing at a stage when the principles of inserting punctuation have been thoroughly grasped and students are ready to take in something new.

1. "Default" means "what the wordprocessor does automatically, unless you give a command to change it" – see the Glossary

Worksheets on saving and printing files should include a note about file names. Point out that a "file" is the name given to a wordprocessor text which you want to save on a disc. Explain the system of naming files which applies to your computer. Students will be modifying texts which you have already saved on to disc, so at the end of a session, their version of the text will be different from the one which you prepared. Explain that there are two ways of naming the files they are working on:

- they can give their version of the file the same name as the original version which you prepared - this has the effect of wiping out the original file and replacing it with the one they have been working on;
- they can give their file a different name - this preserves your original file *and* the student's new version.

(You would be well advised in any case to keep *two* copies of your original files in a safe place, and *never* allow students to work on the only existing copy of anything - or to do so yourself, for that matter!)

Provide a separate worksheet for operating your printer. Make sure that the printer is prepared in advance so that all the student needs to do is to put paper in the printer, switch on and issue the print command from the wordprocessor.

Pre-computer work:

- if language-experience writing is being used, then this will need to be corrected in discussion with students before the punctuation exercise is introduced;
- work on the principles of punctuation will obviously be a prerequisite of this task, but it may also be advisable to prepare orally the text which is to be done on the wordprocessor, or a very similar text;
- students may also need some preparatory work on identifying punctuation symbols on the keyboard, and perhaps on the use of the SHIFT key.

Follow-up exercises:

- either a crib-sheet of the punctuated version of the text for the students to check, or a discussion of the students' own versions - the latter is probably better, because most texts are open to more than one interpretation of punctuation, so there is rarely a "right" answer;

- an obvious follow-up would be to ask the students to review critically other pieces of their own writing, to see where punctuation might have been used to better advantage;
- work on common abbreviations using full stops might be appropriate;
- punctuation work could be used as a lead-in to letter-writing;
- more advanced students could discuss the merits of alternative methods of punctuation – perhaps comparing the modern “fully blocked” style of letter-writing in which punctuation is minimal, with the more old-fashioned dot-and-comma-everything style.

To prepare

- a file consisting of a text stripped of (some of) its punctuation;
- a worksheet explaining how to insert characters into the text (this may not be necessary if “insert” is the default mode, but if the default is “overtyping”, students will need to know how to change to “insert mode”);
- (possibly) a crib-sheet of a correctly-punctuated version of the text;
- a printed version of the uncorrected text for pre-computer discussion;
- worksheets on naming, saving and printing files (unless these were included in the previous task).

NOTES:

In the previous task, students only needed to know about the RETURN, DELETE and cursor keys. Here, punctuation keys are added, and possibly also the use of the SHIFT key.

If you feel that the students can cope with a greater degree of difficulty in the wordprocessing part of the exercise, it may be made more difficult by using lower-case characters throughout, so that part of the task is to swap upper for lower case where necessary. This is in effect a combination of the punctuation exercise with the case-swapping exercise which follows – see the next task for details.

WARNING: *do not underestimate the extra degree of difficulty involved in the “simple” additional activity of swapping upper for lower case!*

TASK 4

Wordprocessor tasks:
case swap
loading files

Language task:
use of upper/lower case

This language task is an extension of the punctuation exercise, and can be used in combination with it.

The task involves inspecting a text which is typed completely in lower case, and swapping letters to their upper case equivalents where necessary. There are at least two ways of doing this, three or even four on some wordprocessors, and the method you use will depend on the object of the exercise and the type of students who will be using it. The methods are:

1. with the wordprocessor in "insert mode", place the cursor immediately *after* the letter which is to be case-swapped, use the DELETE key to erase the lower-case letter, then type in the upper case version, using the SHIFT key;
2. if your wordprocessor has a command (or perhaps a special key) to delete the letter "under" the cursor, place the cursor on the letter to be deleted and give the delete command (or press the appropriate key), then, with the wordprocessor in "insert mode", type in the upper case version, using the SHIFT key;
3. with the wordprocessor in "overtyping mode", place the cursor *on* the letter which is to be case-swapped, and type in the upper case version, using the SHIFT key;
4. if your wordprocessor has a case-swap command, place the cursor *on* the letter to be swapped, and give the case-swap command.

Of the four, Method 1 is the most difficult, and Method 4 the easiest. Bear in mind that only Method 4 requires no knowledge of the keyboard. Method 4 is therefore the easiest one to combine with the previous punctuation task, the other methods needing considerable additional skill and dexterity. Method 3 would be relatively easy on a wordprocessor such as *View*, where "overtyping" is the default-mode, but this would still demand some knowledge of the keyboard.

There are several different ways in which this exercise could be used:

- with beginner readers, it could be used for reinforcing work on the relationship between upper and lower case forms of letters; a text typed entirely in lower case could be changed entirely into upper case, requiring the student to relate the lower-case letters on the screen to their upper-case equivalents on the keyboard (Method 2 would be the most effective in this case. Method 4 would be a bit pointless, being an automatic substitution!);
- with more advanced bilingual and/or literacy students, the exercise could be used for applying the rules of the use of capital letters in English (Method 4 would be the most practical to use, if it exists on your wordprocessor. Method 3 would be satisfactory, provided that the students are not also expected to insert punctuation. Methods 2 or 1 can be used where punctuation is also part of the exercise, so that students will not have to switch back and forth between insert and overtype modes);
- with students who are new to the keyboard, this exercise may be useful as an informal keyboard tutor, helping them to identify and remember the positions of characters on the keyboard (Method 4 would again be pointless for this purpose).
- any or all of the four methods could be used to practise the wordprocessing techniques for their own sake.

LOADING FILES

The students' first job at the machine will be to load the prepared case-swapping text into the wordprocessor and get it on to the screen.

Having learned about saving files in the previous task, students should readily grasp the concept of retrieving files from the disc. The students could practise initially by retrieving the files they saved at the end of the previous session, just to prove that they are still there.

When you prepare a worksheet on loading files, include if necessary an instruction to get the text on to the screen! Some wordprocessors have two separate processes for loading files and transferring them to the screen, others combine the operations.

Pre-computer work:

- beginner readers may need considerable preparatory work on letter formation before approaching the keyboard. For them, case-swapping could well precede the work on paragraphing and punctuation introduced in the previous tasks;

**Follow-up
exercise:**

To prepare:

- more advanced students could write a short language-experience text, and correct it with the tutor. The text could then be used as the basis for a case-swapping exercise. (The alternative is for the tutor to select a text which illustrates a variety of different circumstances in which upper case is used in English);
- using a printed version of the uncorrected text, students discuss the principles of the use of upper case, and decide what alterations need to be made to the text.
- students check the alterations they have made (using printouts of their work) against a corrected master version of the text.
- a file consisting of a text suitable for case-swapping;
- a worksheet on loading files – including getting the text on to the screen;
- a worksheet on case swapping;
- a printout of the uncorrected text for discussion;
- a printout of the corrected text for checking.

NOTES:

A major reason for using a wordprocessor is to make life easier! It makes sense, therefore, to use the easiest method you can find for whatever wordprocessing job you have to do.

Very often, wordprocessors provide more than one method of performing the same job – so, for case-swapping there can be four different ways of doing the job. The point of this apparent overkill is that different methods are appropriate in different circumstances. Your choice of case-swapping technique may well depend on whether you happen to be in overtyping mode or insert mode when you need to make a swap. It is therefore an advantage to know all the methods, if possible.

With students whose main object is the acquisition of language or literacy skills rather than learning wordprocessing for its own sake, you may feel that introducing more than one method would only confuse the issue. How many techniques you bring in at this stage will therefore depend on how interested the students are in the nitty-gritty of wordprocessing, and how much new information you feel they can take in.

TASK 5

Wordprocessor task:
whole-word deletion

Language task:
parts of speech

As with case-swapping, there are several different ways of achieving deletion of text using a wordprocessor. This exercise involves deleting whole words, which can usually be done with a single command.

Prepare a text with selected words removed and replaced with the name of the appropriate part of speech. For example:

Yesterday, my brother and I went to the NOUN to see the NOUN. It was a(n) ADJECTIVE day, and the weather was ADJECTIVE. As we saw the NOUN coming ADVERB towards us, we VERB ADVERB.

(An alternative idea might be to use words which patently do not make sense in the context, and ask the students to identify the nonsense-words). The parts of speech can, of course, be selected to suit the group for whom the exercise is intended. The task is to delete the named parts of speech and insert suitable vocabulary to build up a meaningful text. The students work on the exercise in pairs, completing one set of vocabulary and printing out the resultant text, then working on a new set.

The task can be simplified by allowing students to use any vocabulary that makes sense, so that the texts generated may be totally different in content and meaning, though using the same structural skeleton; this could be used to encourage vocabulary-extension and attention to spelling. Another possibility would be for the tutor to provide the text with a title, thus limiting the area of acceptable vocabulary. Alternatively, there could be a stipulation that every version of the text must be essentially the same in meaning, forming the basis of a discussion of register and/or style, although the text would need to be very carefully chosen for this to work well.

Pre-computer work:

- preparatory work on identifying and classifying parts of speech;

**Follow-up
exercises:**

- oral group work, making substitutions for selected parts of speech in known texts (possibly their own language-experience writing);
- discussion of a printout of the wordprocessor text, with suggestions for possible substitutions. If the students can cope with an added language task, asterisks or some other symbol could be used in the text instead of naming the missing words, so that the first job would be to identify what kinds of words were needed to fill the gaps.

To prepare:

- could include printing out the texts and comparing the effects of different substitutions.
- a file consisting of a text marked with suitable parts of speech;
- a worksheet describing the word-delete function (see *NOTE (b)* below);
- a printout of the exercise for prior discussion, if required.

NOTES:

(a) *There are many ways of deleting text from a wordprocessor file, and your wordprocessor should provide at least some of the following:*

- delete a character to the left of the cursor;
- delete the line in which the cursor is sitting;
- delete a character at the cursor position;
- delete a sentence;
- delete a word to the right of the cursor;
- delete a paragraph;

If your wordprocessor lacks some of these commands, the same effects can be achieved by combining one or more of the commands you have at your disposal. It is useful to learn as many of these as you can, so that text-handling can be carried out as economically as possible. You may find that your wordprocessor offers still more ways of deleting that are not mentioned here.

(b) *Note that some wordprocessors will operate word-delete with the cursor at any position in the word, while others only delete forward from the cursor position. If yours is the kind that deletes forward from the cursor, be sure to stress in your worksheet the importance of placing the cursor at the start of the word to be deleted.*

TASK 6

Wordprocessor task:
whole-word deletion

Language task:
multiple choice – structure
and/or vocabulary

The wordprocessor task is similar to the previous task, but there may be added complications depending on the wordprocessor you are using, so you would be well advised to read the *NOTES* below before writing your text.

Prepare a piece of continuous text containing a number of multiple-choice options, on these lines.

The man who <come> <comes> <came> into the room said "Come and <see>
<watch> television with me. There's a programme on about <cook> <cooking>
Chinese food – I'm sure you <will> <shall> enjoy it!"

The student's job is to move the cursor to one of the *wrong* choices and give the word-delete command to eliminate that word. The procedure is then repeated until all the wrong choices have been deleted. It may be necessary to "reformat" the text before printing, as some wordprocessors (for example, *Wordstar*) do not automatically reorganise the lines when words are deleted from the text. When the task is finished, the students can print out copies of their final version to take away.

Remember that once the words have been eliminated they will not be retrievable, so when the exercise is finished, what is left will be a version of the text which may well be incorrect, with no means of recalling what the other options were. The students should therefore be provided with a printed copy of the original text, or alternatively you could put two copies of the text into the wordprocessor file, so that the student can correct one and use the other for reference. This way, when the file is printed, the original will appear on the printout with the student's work, for comparison.

Pre-computer work:

- appropriate vocabulary work preparatory to the multiple-choice exercise;
- group discussion and preparation of the exercise.

Follow-up exercises:

- group comparison of printouts of the students' final versions, to thrash out any difficulties;
- discussion of any word-groups which the students themselves find confusing, for reasons of spelling, meaning or structure.

To prepare:

- a file consisting of a multiple-choice text typed in a format which your wordprocessor's word-delete function can handle (see *NOTES* below);
- a worksheet explaining the word-delete function, if the details of this are different from the previous exercises;
- a printout of the exercise for prior discussion, if required.

Choose the right words and DELETE the others:

The man who|come/comes/came into the room said "Come and watch television with me. There's a programme on about cook/cooking Chinese food - I'm sure you will/shall enjoy it!"

When you are ready, scroll the screen up to check your answers.

Press ESC if you want to give a command.

Fig. 2. The form in which multiple-choice is presented depends on the wordprocessor you use. *Oxfordshire Write* (RM Nimbus).

NOTES

Be careful when constructing a multiple-choice exercise that your wordprocessor's word-delete function will operate correctly on the text you are preparing.

Some wordprocessors operate word-delete only on alphabetic characters, while others will delete non-alphabetic characters too. If yours deletes only alphabetic characters, then you can arrange your multiple-choice elements without intervening spaces, like this:

this/that/these

Placing the cursor at the start of any one of the words will delete only that word, leaving the other words (and the markers) intact. The markers will have to be deleted separately. If on the other hand your wordprocessor deletes non-alphabetic characters along with the alphabetic ones, then you must take care to leave spaces between the multiple choice elements, like this:

<this> <that> <these>

Whichever form you choose, be sure to explain in your worksheet exactly what steps must be taken to delete both words and markers to leave a "clean" piece of text.

TASK 7

Wordprocessor task:
consolidation: cursor movement
and deletion/insertion

Language task:
proofreading for elementary
errors

Prepare a file consisting of a short paragraph – three or four lines – containing a dozen or so typing errors (that is, errors that a typist might make, rather than spelling mistakes per se). Include only errors which the students may reasonably be expected to recognise. You may be able to find a suitable piece of a student's own writing, or – if you are reluctant to expose students to "deliberate mistakes" – you could introduce "nonsense errors", using characters which do not normally appear in English words.

Try to include at least one example of each of the following types of error:

omission	– e.g.	"realy" for "really"
addition	–	"comming" for "coming"
duplication	–	"not not" for "not"
metathesis	–	"yuo" for "you"
missing word	–	"been" for "has been"
case	–	"john" for "John"

Correcting these errors will use a full range of cursor movements, plus deletion and insertion of letters and words, and case-swapping. This does not involve any new wordprocessing techniques, but provides a chance to consolidate what has been learned so far. If you feel that the students are fairly confident, you could include the command to "delete the character at the cursor position", which is a useful addition to the range of methods of deleting text.

**Pre-computer
work:**

- use a printout of the uncorrected text to identify with students what errors need to be corrected, and what the corrections are.

**Follow-up
exercises:**

- if necessary, follow up with work on specific spelling problems arising from discussion of the text;

- away from the computer, students may like to look at some handwritten work (perhaps their own language-experience writing) to discuss what could be done to correct it on a wordprocessor. This should encourage a critical faculty when re-reading work, and a gradual acceptance that mistakes are not irretrievable.

To prepare:

- a wordprocessor text containing errors as described above;
- a printout of the uncorrected text for prior discussion;
- a printout of a corrected version of the text (but look at the following two tasks before preparing this);
- possibly a single sheet summarising the editing techniques covered in previous tasks: cursor movement, DELETE and RETURN, insert mode, word-delete, and adding "delete character at cursor position" if you have decided to use it.

NOTES:

This task practises basic editing techniques which are crucial to an efficient use of the wordprocessor, and which should be mastered before proceeding to more advanced techniques. (The following two tasks provide further practice).

Although it may take only a few seconds to learn a wordprocessor command, it takes considerable practice at the keyboard to "internalise" the command so that it becomes second nature. This is especially true when sessions at the wordprocessor are few and far between, so these consolidation exercises should not be shunned in favour of progressing to something new.

Accomplished users of the wordprocessor, however brilliant their typing, still use deletion and insertion for correction more than any other feature of the wordprocessor!

TASK 8

Wordprocessor task:
consolidation: cursor movement
and deletion/insertion

Language task:
proofreading from a fair
copy

Following the correction of text as described in the previous task, give the students a short dictation directly on to the wordprocessor (perhaps continuing the theme or story of the previous text), stressing that mistakes should *not* be corrected during the dictation, and that – for the purpose of this exercise – they should feel free to make as many mistakes as they like, even deliberate ones! This can produce great hilarity, and the students should not feel threatened in any way – explain that you are using the dictation solely for the purpose of practising keyboard editing, and that it is not a test of any skill. Then provide a handout containing a correct version of the dictation, so that students can correct their work.

Ask the students to name, save and print their corrected file.

Pre-computer work:

- if you feel it is advisable, advance preparation of the dictation material.

Follow-up exercise:

- more detailed examination of the corrected text.

To prepare:

- a dictation;
- a handout containing a correct version of the dictation.

NOTES:

This exercise should be used as a little light relief. It is quite good fun trying to make mistakes!

When the text has been corrected on screen, use a printout for the final checking process, as it will be easier to spot errors on the printed page.

TASK 9

Wordprocessor tasks:
copy typing
consolidation: cursor movement
and deletion/insertion

Language tasks:
copying
proofreading

This is the first exercise in which the students are required to type a text. While they are not at this stage expected to compose at the keyboard, nevertheless using the full keyboard is a considerable advance on the skills called for in previous tasks, and allowance should be made for this when timing the exercise, especially for those who are unfamiliar with the QWERTY layout of the keys.

Provide the students with a printed version of a short text (perhaps a continuation of the texts used in the previous two tasks), or a piece of their own writing which you have already corrected with them (a typed *not* a handwritten version), and ask them to type it on the wordprocessor, encouraging them to ignore any mistakes they may make. Then ask them to name, save and print the text.

Using the printout of their typing, they can then proofread it against the original, mark any differences, then go back to the wordprocessor to correct the typing errors.

Pre-computer work:

- sample proofreading exercises (on paper, not on the wordprocessor), comparing two printed copies of a text and picking out differences between them. You could perhaps use a text altered in such a way that some of the differences were valid alternatives to the original, while others were genuine errors which needed correction, asking the students to identify the two types;
- possibly correcting students' own writing with them for use in this exercise (the tutor will then have to wordprocess and print this for the purposes of this task).

Follow-up exercise:

- typing samples of handwritten text, and proofreading the printouts against the original. This is a considerable advance in difficulty compared with copy-typing from a typed text, especially if the handwriting is not the student's own!

To prepare:

- a printed version of a passage to copy-type. (A single handout could possibly contain all three passages for this exercise and for the two previous ones – a corrected version of the original file, a corrected version of the dictation, and a typing passage. This is especially appropriate if the three passages form part of a continuous text).
- other pairs of texts for proofreading, both printed and handwritten.

NOTES

Proofreading from the screen is extremely difficult. Even experienced typists will miss errors on the screen which they would readily pick up on paper. All proofreading exercises should therefore be done from paper copies.

Another reason for doing this is simply to give the eyes a rest from the screen!

TASK 10

Wordprocessor tasks:
consolidation: cursor movement
and deletion/insertion (deletion of phrases)

Language task:
refining text – tautology

Prepare a text containing examples of tautology, such as:

Repeatedly and time and time again, first-time newcomers to information technology remark and comment on the ease and facility with which wordprocessing can be learned. All the boring, tedious aspects of writing can be eliminated and totally overcome by the use of the wordprocessor.

Discuss the text with the students, and decide where the language can be "pruned" to say the same thing more economically. Ask the students to load the file, make the alterations, save the text under a new name, and print it.

Pre-computer work:

- collection and discussion of vocabulary with broadly similar meanings; this could be done by examining a text known to the students and finding alternative ways of expressing its content, or by looking for phrases on a theme (eg. time – sooner or later, in due course, in the course of time); the discussion could be broadened to include the stylistic or semantic effects of various choices;
- discussion of the wordprocessor text.

Follow-up exercise:

- investigation of some of the students' own writing, to see whether this suffers from tautology, or whether a different way of saying the same thing would be a stylistic improvement.

To prepare:

- a wordprocessor text containing suitable examples of tautology.

NOTES:

This could be simply further consolidation of the editing procedure, but if your wordprocessor makes provision for "deletion of phrases" (as distinct from deleting single words or deleting whole lines) then this would be a good time to introduce the command.

*Some wordprocessors allow the deletion of phrases by "marking" the phrase, then deleting it with a single command, rather than deleting it a word at a time (or, as is commonly done, by simply keeping your finger on the DELETE key!). In **Word**, for example, on the Nimbus computer, marking any selection of text is very simply done, and the deletion is accomplished with a single keypress. With others however (notably **Wordwise** on the BBC micro) the process of marking text is so cumbersome that it is faster and much less trouble to use the "delete whole word" command the requisite number of times!*

TASK 11

Wordprocessor task:
overtyping/insert

Language task:
cloze

There are two ways of preparing cloze exercises for students to complete on the wordprocessor. Which one you use will depend partly on the students' prowess with the wordprocessor, but mainly on the degree of language difficulty which they can cope with.

**Overtyping
method:**

This method is the simpler one in terms of language. In this version, every clozed word will be replaced by the appropriate number of symbols on the screen, like this:

When John caught the *** home, the conductor asked him for his ****, and he suddenly ***** that someone had picked his *****. He had no ***** at all.

Here the student has a visual clue to the length of the missing word, as well as contextual clues. There is little scope, however, for imaginative efforts!

If you are unfamiliar with overtyping, it would probably be a good idea to read the more detailed notes on the overtype mode in the next task before embarking on this exercise.

The student puts the wordprocessor into overtype mode, places the cursor on the first symbol of one of the clozed words, and types the word over the top of the symbols. If mistakes are made, they should be corrected by re-positioning the cursor and typing over the mistakes, *not* by using the DELETE key.

**Insert
method:**

Using insert mode, the passage is typed using one symbol only to replace each clozed word:

When John caught the ★ home, the conductor asked him for his ★, and he suddenly ★
that someone had picked his ★. He had no ★ at all.

In this way, there is no visual clue as to the length of the missing word, and it may be possible to complete some of the gaps in more than one way

The student puts the wordprocessor into insert mode, then *either* places the cursor on a replacement symbol, uses the "delete whole word" command to erase the symbol, and inserts the missing word, *or* places the cursor immediately after the symbol, erases the symbol with the DELETE key, and inserts the missing word

Remember that if the method you decide on is not the default mode of your wordprocessor, you will have to include in your worksheet a reminder of how to switch from one mode to the other, and how to switch back again when the exercise is finished.

**Pre-computer
work:**

- any vocabulary and/or spelling preparation that is necessary,
- if necessary, prepare the actual text in advance

**Follow-up
exercises:**

- checking a printout of the text for errors of vocabulary or spelling,
- reinforcement of vocabulary or spelling at a later stage

To prepare:

- a text clozed in one of the ways described above,
- a worksheet explaining the use of overtype or insert (whichever you have chosen), including how to switch from one mode to the other if necessary.

NOTES

One advantage of doing cloze in this way is that you get a printout of the correct version at the end. Another advantage when students are working in pairs or small groups is that the vertical screen makes co-operation on the text much easier.

*It is also extremely easy to prepare several different versions of the same cloze text (perhaps clozing the nouns on one occasion, the verbs the next time, then prepositions, and finally withdrawing almost all the words). This means that students get a chance to work on the text in depth. It is also possible to use this technique as a build-up to using a program such as **Developing Tray**, where all the support is withdrawn, and students recreate a text using only the barest minimum of support.*

If you prepare several versions of the same text, it might be an interesting exercise to let different groups of students work simultaneously on different versions, then compare notes on the comparative difficulty of finding replacements for, say, nouns in one version and prepositions in another.

TASK 12

Wordprocessor task:
overtyping

Language tasks:
form-filling, "audio-typing", the
language of questions and answers

Prepare a file consisting of a blank form on which the students will type information about themselves (or somebody else). The form should be laid out exactly as it would be on a printed sheet, with sufficient space for students to type in the data to complete the form. It can include the usual questions found on forms – name, address, telephone number, etc., – and perhaps one which would require an extended answer – for example, "Give further details below".

There are several ways of using the form, depending on your objectives for the lesson. The simplest way would be to ask the students to complete it with information about themselves (or about an imaginary person, if you feel the information requested might be sensitive). You could have several different examples of forms for different purposes (medical, DHSS, tax, enrolment etc.) and repeat the exercise in different ways over a period of time. An alternative way of using the form is to provide students with a biographical sketch of a fictitious person, containing all the necessary information, but not necessarily in the right order.

Another possibility is to get students working in pairs at the machine. One student should be the typist, and the machine should be angled so that only the typist can see the contents of the screen. The typist then questions the partner to elicit the information (not necessarily genuine!) required to complete the form on the screen. This may include asking the partner to confirm spellings, for example. The typist fills in the form on behalf of the partner, who prints out the completed form and checks it for accuracy. The pair then change roles.

The above method is "audiotyping" in the sense that the typist is typing information which has only been heard and not seen, but she will also be able to question her partner if clarification is needed. It could be turned into a genuine audiotyping exercise by pre-recording the information on a cassette. This method would only work, however, if you have sufficient equipment to allow each student a machine to work at; you would probably also need earphones for the cassette recorders, if a number of students were doing the same task.

The wordprocessor task requires the use of the "overtyping mode". If you have already done Task 3, you will know whether your wordprocessor's default mode is "insert" or "overtyping". (Refer back to Task 3 if you need to refresh your memory as to the meaning of these terms). If the default mode is "insert", you will need to include in your worksheet the command to switch to "overtyping mode". If you are using *Interword*, you will not be able to do this activity in the form in which it is described here, as *Interword* has no overtyping mode.

Overtyping mode is needed so that students can type information on the screen without disturbing the layout of the form.

Stress to students that they should only type into the blank spaces on the form, otherwise the form itself will disappear! It would probably be helpful to mark the spaces for answers with square brackets, and warn students that they must type only inside the square brackets. For example,

Family name: [] First name(s): []

Be sure that the space within the square brackets is sufficient for the longest entry that may be required. If you have an extended-entry question, place it at the end of the form, without brackets, where it will not affect the layout, and can be as long as the student needs to make it.

Also point out that students should not attempt to DELETE their errors in the usual ways (again, this would interfere with the layout of the form), but should use the cursor keys to reposition the cursor at the start of the mistake and type over it.

Students could be encouraged to complete the form in capital letters, as BLOCK CAPITALS are often requested on official documents. Explain the use of the CAPS LOCK key, if this is the first time they have met it.

**Pre-computer
work:**

- (if the exercise is to be done in pairs) preparation of the questions that may need to be asked to elicit the information to complete the form, including spelling out names and addresses;
- preparatory work on the language of forms, perhaps including the different ways of expressing dates and times.

APPLICATION FORM

Family name: [] ; First names: []

Address: (street) []
 (town) []
 (postcode) []

Sex: () Age: () Marital status: ()

Date of birth: (day) () (month) () (year) ()

Education: (type full details of your education below)

COMMAND: Alpha Copy Delete Format Gallery Help Insert Jump Library
 Options Print Quit Replace Search Transfer Undo Window
 Edit document or press Esc to use menu
 Page 1 () Microsoft Word: FORM.DOC

Fig. 3. A sample form-filling exercise. Word (RM Nimbus).

Follow-up exercises:

- handwritten work on real forms, perhaps again using the same technique of eliciting information from another person to complete the form;
- discussion of the kinds of forms we are asked to complete, what happens to the information we supply on those forms, and what rights we have over information stored about us – particularly information kept on a computer (controlled by the Data Protection Act).

To prepare:

- files containing "model" forms of various types, suitable for "overtyping" by the students;
- a worksheet reminding students
 - to enter "overtyping mode" (if necessary);
 - to type only in the blank spaces on the form;
 - not to DELETE their errors but to OVERTYPE them;
 - to reset "insert mode" on finishing the exercise.

NOTES:

Students are bound to make mistakes with "overtyping mode" until they grasp the principles.

The most common source of difficulty is the temptation to use the DELETE key after making an error, and this can have disastrous effects! If the form is seriously damaged, it is probably better to cut your losses and reload the blank form from the disc, even if this means losing your earlier work. If the layout is only disturbed in a minor way, it is possible to reverse the damage by returning to insert mode, continuing with the exercise carefully and slowly until the form restores itself to its original layout, and then going back to overtype mode again to complete the exercise.

This is one of those things that sounds extremely difficult to cope with, but in practice, once you grasp the principles, is very easy to achieve!

If your students need to do a lot of work on form-filling, it would be easier to set up the blank forms on a database rather than on a wordprocessor. In this way, the actual completion of forms would be much easier than with the wordprocessor and the completed forms could be saved and used for preparatory work by future generations of students.

TASK 13

Wordprocessor tasks:
using tabs
typing a table

Language task:
vocabulary extension

Tabs are very useful when you are working with words (or numbers) which need to be arranged neatly in columns.

Tabs on a wordprocessor are very similar to tabs on an ordinary typewriter: each "tab-stop" marks the beginning of a new column of text, and the tab-stops are set at points along the width of the screen so that the next stop can be reached with a single key-press, instead of repeated use of the space bar. This makes lining up columns of text a great deal easier and faster.

If you are unfamiliar with the use of tabs, you should read the introduction to Task 14, which explains their use in some detail. You will also need to read the *NOTES* below about potential pitfalls before deciding whether or not to do this task with your students.

This task can involve considerable work away from the computer in advance of the wordprocessing part of the task. Choose an area of vocabulary which will interest and benefit your students, and which lends itself to a tabular form. One example of this might be vocabulary related to animals:

GENERIC	MALE	FEMALE	YOUNG	MEAT	NOISE
cow	bull	cow	calf	beef	low
sheep	ram	ewe	lamb	veal	bleat
pig	boar	sow	piglet	lamb	
				mutton	
				pork	grunt
				ham	
				bacon	

Another example could be the kind of word-families dealt with in Task 24.

Ask students to prepare lists of vocabulary on the chosen topic to type on the wordprocessor as tables, like the example above. It would be wise to limit tables to text rather than numbers, as tabulating numbers brings in extra complications, especially if the numbers contain decimal points. (Many wordprocessors have a special tabbing feature to deal with decimals, but there is little point in introducing such a feature at this stage).

Spend some time planning the layout of the table on paper before going to the wordprocessor, using the principles outlined in the introduction to Task 14, then *demonstrate* how to clear and set tab-stops, and how to use the tab key, before asking the students to type their tables. To make the exercise easier, clear *all* the pre-set tab-stops and set new ones to suit the table that is to be typed

Pre-computer work:

- preparation of vocabulary;
- examination of model tables (on paper) to see how they are laid out;
- thorough discussion of the layout of the students' table(s), including positioning of the tab-stops.

Follow-up exercise:

- use of the vocabulary in creative or guided writing.

To prepare:

- a worksheet containing all the necessary information about how to clear, set and use tabs (see Task 14 for ideas about this);
- some model tables to examine prior to planning the students' own tables.

NOTES

Some wordprocessors handle tabbing better than others, but most have idiosyncrasies which you need to know about before deciding to inflict them on your students.

It is useful for tabbing purposes to have a WYSIWYG wordprocessor – that is, the tabbing effect that you see on the screen is what you will see on the paper. View (BBC micro) is not WYSIWYG in many other respects, but it is quite good for the purpose of tabbing. When you press the tab key, the cursor moves across the screen to the next tab position, and it is easy to see where each column of text will start.

Wordwise (BBC micro) on the other hand merely shows that you have pressed the tab key by printing an arrow on the screen, and there is no indication of how the finished product will look until you enter "preview" mode. This means that you need to

- clear all the pre-set tab-stops and use only the ones you need;
- prepare what you are doing very carefully before going to the wordprocessor;
- use the preview facility often while you are typing in order check that the tabs are in the right positions.

Another problem with **Wordwise** is the fact that you are forced to type on a 40-column screen, whereas your printing will mostly be done on A4 paper, which is nominally 80 columns wide. This means that every line that you print on paper occupies two lines on the screen, which makes tabbing doubly difficult.

```
Words-28      Characters free-26721 I

GENERIC→MALE→FEMALE→YOUNG→MEAT→NOISE
cow→bull→cow→calf→beef→low
→→→→veal
sheep→ram→ewe→lamb→lamb→bleat
→→→→mutton
pig→boar→sow→piglet→pork→grunt
→→→→ham
→→→→bacon
```

Fig. 4. Tabbing is very difficult on a non-WYSIWYG wordprocessor, as it is hard to visualise how the printout will look. *Wordwise Plus* (BBC).

Word (RML Nimbus and other MS-DOS machines), though WYSIWYG in most respects, has some very bad habits when it comes to tabbing, and the best way of using tabs with this wordprocessor is to clear all the pre-set stops and use only your own, set using a style sheet (a feature which allows you to add your own commands to the **Word** wordprocessor).

Whichever wordprocessor you are using, one thing is worth bearing in mind: once you have got the tabs sorted out, save the file, so that you can go back to it if disaster strikes. Then, as you add new material which you are satisfied with, save the file often, as this can avoid a lot of frustration should anything go wrong at a later stage.

GENERIC	MALE	FEMALE	YOUNG	MEAT	NOISE
cow	bull	cow	calf	beef	low
sheep	ram	ewe	lamb	veal	bleat
pig	boar	sow	piglet	mutton	grunt
				pork	
				ham	
				bacon	

Fig. 5. A WYSIWYG wordprocessor makes tabbing easy. *Interword* (BBC).
The printouts from Figs. 4 and 5 would be identical.

It is very disheartening to put in a lot of work, only to lose or spoil everything you have done, and the "save little and often" rule is worth remembering – not only for tabs, but for any wordprocessing job which has cost you dearly in effort. Don't wait until you lose three or four hours' typing to discover this the hard way!

TASK 14

Wordprocessor task:
using tabs

Language task:
writing a C.V.

Preparing a curriculum vitae is a job that many students will want to tackle at some stage, and this is one wordprocessor exercise where the printout can serve a very practical purpose. Another advantage of preparing a C.V. on the wordprocessor is that it can easily be modified without having to be retyped.

When the output is going to be used for an important purpose, there is an added incentive to make the product attractive. The use of tabbing can make this goal easier to achieve (but read the *NOTES* to Task 13 before deciding to do this task with your students).

There are usually default tab-stops on a wordprocessor, often pre-set every ten spaces along the width of the screen, so there will be tab-stops set at 10, 20, 30, 40 characters, etc. This means that you can tab across to a position 30 characters along the width of the screen, simply by pressing the tab key three times. You need to know which is the tab key, of course, and to know what the default tab-settings are on your wordprocessor.

If you are lucky, the default tabs may be suitable for the columns you want to lay out, but you will probably need to redefine some or all of them for your purposes. This means learning how to "clear" the tab-stops you do not want, and to "set" tab-stops in the places where you need them. Tab-stops can be cleared individually, but there is often a command which will allow you to clear all stops at one fell swoop.

Your worksheet for students will therefore need to cover:

- what tabbing is, and how to achieve it;
- which are the pre-set tab-stops on your wordprocessor;
- how to clear some or all of the tab-stops;
- how to set new tab-stops to suit the work you are planning to do.

Students will need to remember when setting tabs that a new tab-stop should allow space before it for the longest possible entry in the previous column of text, plus a reasonable gap between columns. For example¹, if the first column is a date in the form "17.10.82", the next tab-stop must be set at, say, position 11 to allow space for the date to be typed plus a couple of spaces as a gap between columns.

0	1	2	3	4	5	6
1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890
^	^					

17.10.82 This is the next column, starting at position 11....

This example is easy, because the maximum length of the date is easy to determine, but if the width of a column is variable (for example, job details), space must be allowed for the longest possible entry you may need to make in that column:

0	1	2	3	4	5	6
1234567890	1234567890	1234567890	1234567890	1234567890	1234567890	1234567890
^	^		^		^	

17.10.82	Duogate Dairies	van-driver
25.04.83	Empirical Chemicals	cleaner
01.01.85	Safebury's	shelf-filler

The second and third columns have variable-length entries, so the third column must begin at position 32 (not 28, where you may be tempted to put it after typing "Duogate Dairies"), and a fourth column would begin at position 46, to allow space for the "shelf-filler" entry in the third row.

1. In the examples, the rows of numbers indicate character positions on the screen – the top row indicating tens and the second row showing units. The third row of arrowheads shows the tab positions.

This means, of course, considerable advance planning of the data which is to go into the C.V., and meticulous attention to detail. An advantage of the wordprocessor in this respect is that, if there is a mistake in the layout on first printing, it is very easy to edit the tab-stops without having to scrap the whole document.

One tip worth bearing in mind is that it is usually better to clear *all* the tab-stops you don't need, in order to avoid confusion – especially where, for example, the pre-set stops come very close to the ones you wish to set yourself.

Different wordprocessors handle tabbing in different ways, and before you decide to embark on this exercise with your students, it would be worth wordprocessing your own C.V. to see what problems they will encounter.

Pre-computer work:

- discussion of the type of information a C.V. should contain, and how to organise it on the page;
- ask the students to make brief notes of the things they have learned to do in the last five (or ten) years; although this produces groans of disbelief at first, it is surprising how many achievements can be listed – driving, cooking, speaking English, sewing, taking up a new sport, doing a course (even if it is not certificated) and so on. Apart from being an ego-booster as students realise how much they have accomplished, this exercise may uncover things which would be appropriate for adding to a C.V. and which may otherwise have been forgotten;
- drafting the C.V. on paper, with special attention to the widths of columns; this is one exercise where advance planning with pencil and paper can pay dividends.

Follow-up exercises:

- examining a printout of the first draft, to see whether changes in layout or language would be advisable;
- drafting a model letter such as might be suitable to send with a C.V. for a job application. (This could be linked with the next task).

To prepare:

- possibly model C.V.s and model letters of application for discussion;
- a worksheet containing all the necessary information about tabbing, and clearing and setting tabs.

NOTES:

Learning wordprocessing can be interesting but, like learning a language, the main object is to get to a stage where you can do something useful with what you have learned. In successfully completing this activity, students should gain a sense of achievement – plus a piece of work which they will be able to use for their own benefit.

TASK 15

Wordprocessor tasks:
tabs,
paragraph indents
justification/ragged right

Language task:
composing and typing a
formal letter

Students are provided with a letter which needs a reply. The nature of the letter will depend on the needs of the students who are to do the exercise, but it would be useful to give them a letter similar in appearance and register to the one which they will be sending in reply.

The main emphasis in the wordprocessing part of this exercise will be on the layout of the letter. Students can use their experience of tabbing from the previous tasks to position the date on the righthand side of the page (or perhaps learn a new command for this, if one exists on your wordprocessor – as on *Wordwise +* and *Word*), and for the “signing off” lines at the end. They may also like to indent the start of each paragraph with a short tab (some wordprocessors – like *Wordwise* – have a special “temporary indent” command for this purpose). However, the modern fashion is for “fully blocked” paragraphs (i.e. the first line of the paragraph starting at the left margin) so if students are likely to be interested in developing secretarial skills professionally, they should not be encouraged to indent paragraphs!

Another point for consideration is whether or not to “justify” a letter. “Justification” makes the righthand margin of your text into a straight line, the same as the lefthand margin. When you write a letter by hand (or type it on a typewriter) the lines are of different lengths, as they are in this paragraph – an effect called “ragged right” – but the wordprocessor can introduce extra spaces between words so that the righthand margin is “justified”, as it would be in a book or newspaper. Many people prefer to use ragged right for letters, because it seems more “personal”, but as wordprocessing becomes accepted as a part of everyday life, justified right margins are being used increasingly for letters.

Some wordprocessors have justification as a default, but with others the default is ragged right. You will need to check which is the normal mode on your wordprocessor, and include the command to change the default on your worksheet if necessary.

Pre-computer work:

- look at examples of formal letters of various kinds and discuss them from the point of view of appearance and the style of language used;
- introduce orally the subject-matter of the letter which is to be answered, and ask the students (working in pairs) to role-play the conversation which might take place if the letter-writer and the recipient met to discuss the subject instead of exchanging letters on it; this should produce ideas about the content of the written reply;
- read the letter which is to be answered, and discuss what needs to be included in the reply and the form of words which might be used;
- compare the forms of words which you would write with the words used in the role-play;
- students could be asked to draft a reply by hand, but it is probably more useful at this stage to practise composition at the keyboard, to encourage flexibility in composition skills, and the skills of re-reading and revision.

Follow-up exercises:

- discussion of the replies, examining their suitability in terms of content and style; students' efforts could perhaps be compared with a model reply;
- the initial exercise could be built up as part of integrated work on applying for a job, linking it with the previous task on writing a C.V. For example:
 - pairs of students construct job advertisements, and print them out (this in itself provides material for a range of work based on local newspapers, including interpretation of the abbreviations used in adverts and their conventions);
 - each pair chooses an advertisement written by another pair, and writes a letter asking for a job description and further details;
 - the students who wrote the advertisement then construct a sheet of information for the prospective applicants (this stage may be by-passed if required);
 - using what they can glean from the original advertisement (and the information sheet if this is supplied) pairs then write C.V.s, plus a letter of application for their selected advertisement;
 - the activity might be finished off by role-playing interviews for the jobs.

To prepare:

- a letter to which students will reply;
- a sheet containing a reminder about tabbing, instructions on how to indent the start of a paragraph (if this is to be done) and how to switch justification on and off;
- (for the integrated exercise) job advertisements to be discussed, and if possible job descriptions.

NOTES:

At this stage of learning wordprocessing, the basic techniques of editing have been mastered, and there is much more opportunity to concentrate on the language element of the task, and start to use the wordprocessor as a tool. In the integrated activity suggested above, the wordprocessor is in fact a small element in the whole language-acquisition process, but nevertheless it is an important element.

The stage at which students begin to say "I wrote a letter today" instead of "I used the wordprocessor today" is the stage when wordprocessing becomes a real benefit in the classroom.

TASK 16

Wordprocessor tasks:
tabs
hanging indents

Language task:
paragraphing

A "hanging-indent" allows a paragraph number to appear flush with the left margin, while the text of the paragraph is indented, as in the following example:

The committee discussed four major issues:

1. The provision of crèches at all the sites where there was sufficient demand.
2. The establishment of twilight classes to serve parts of the community which have so far been unable to make use of existing provision.
3. An increase in the number of outreach projects.
4. The proposed budgetary cut of 10% for the coming year, which makes all the above impossible.

The language task involves reading a suitable text which has been typed as a continuous piece, and deciding how it can best be divided into numbered paragraphs. The text may have been previously typed by the tutor into a wordprocessor file, so that the student's job is to break the text into paragraphs and add hanging indent commands and paragraph numbers. Alternatively, the students may be presented with a paper copy of the unformatted text and asked to type it on the wordprocessor, formatting and numbering as they type. They can then print out a copy of the finished version.

Hanging indents can often be achieved with a single wordprocessor command, and are sometimes related to tab-stops, in that the first line of each paragraph starts flush with the left margin, and subsequent lines are indented as far as the first tab-stop. With the *Wordstar* wordprocessor, the command CTRL OG can be used once to indent to the first tab stop, twice to get to the second tab stop, and so on. *Word* allows you to set up for yourself exactly how you want the hanging indent to appear. *Wordwise* lets you fool the printer into making a hanging indent by issuing an ordinary indent command halfway along the first line of the paragraph (that is, too late for the printer to act on the current line, but in time for all subsequent lines to be indented).

Simple wordprocessors may not have a separate hanging indent command, and in this case, you will have to format the text in the same way as you would do it on a typewriter – that is, use the RETURN key to end each line, and set a suitable tab stop so that pressing TAB once will indent the next line to the correct position. However, you will not have a justified right margin if you use this method.

Pre-computer work:

- introduce the students to the concept of ordering paragraphs by asking them to sequence a text in which the indented paragraphs have been jumbled. Instead of using numbers for the paragraphs, a symbol such as an asterisk could mark the left margin. The paragraph-sequence should be readily identifiable without numbering, of course,
- follow this up with a less obvious text, again with numbering omitted, and ask students for their opinions on how it might be logically ordered (there may be more than one solution)

Follow-up exercises:

- discuss a printed text (say, the minutes of a meeting) where there are paragraphs and sub-paragraphs. Observe how these are "nested" (each new set of sub-paragraphs being indented further than the "parent" set on which it is dependent), and how different sets often use different labels so that they can be readily identified (for example, 1/2/3 – i/ii/iii – 1.1/1.2/1.3 etc.);
- introduce a piece of jargon (say, from an insurance policy- "This applies as in Clause 4, paragraph 2, sub-paragraph 5(iii) unless Clause 3, paragraph 5 is applicable") to show how paragraphing can be taken to such extremes that it clouds the issue rather than clarifying it.

To prepare:

- a worksheet on hanging indents, and a reminder on how to set tab stops, if necessary;
- a suitable text, typed as a single paragraph, and prepared as a handout;
- if required, a wordprocessor file consisting of the same text;
- if required, texts to be used in sequencing exercises, (see Tasks 19 and 20 for details of block move if you want to do this on the wordprocessor);
- a more complex text, including sub-paragraphs at various levels, for discussion.

NOTES:

The wordprocessor skill involved in this task is minimal, but the language work could be quite detailed, including:

*comprehension
categorising information
logic
sequencing
layout*

The wordprocessor adds interest to the rather mundane business of listing information, and allows experimentation with different layouts, plus a final good copy to take away.

TASK 17

Wordprocessor tasks:
tabs
hanging indents

Language task:
writing a questionnaire

The language task involves composing a questionnaire and writing it up on the wordprocessor. If there is time, this could form part of an integrated series of activities (see *Follow-up exercises* below).

Examination of different types of questionnaire will reveal that they invariably use "hanging indents" (see the previous task for details of this technique) to make the questions and multiple-choice answers more readable, as in the following example:

1. How often have you made use of British Telecom's Yellow Pages in the past month?
 - (a) never
 - (b) rarely
 - (c) occasionally
 - (d) frequently

Pre-computer work:

- students look at one or two sample questionnaires – popular magazines often have light-hearted questionnaires, and one of these could be compared with something more serious, such as a census form or a local authority survey; the samples are then discussed from the point of view of layout – what makes them attractive and readable;
- students discuss topics which would be suitable for questionnaires – perhaps starting with a discussion of questionnaires which they have come across themselves, possibly as a result of being questioned by market researchers, or ones they have found in magazines;
- having fixed on a suitable subject for the wordprocessor questionnaire, questions can be formulated orally (perhaps with students taking brief notes) or drafted by hand, before pairs of students move to the wordprocessor to prepare a finished product;

- if there is enough scope in the chosen topic, it should be possible to formulate many more questions than will actually be used in the final questionnaire, so that different groups can make their own selection of, say, five or ten questions, which will add variety to the final printouts.

Follow-up exercises:

- students can compare their questionnaires, then swap and complete each others' versions. At this stage, it should become obvious if any of the questions are ambiguous or difficult to understand, and this can form the basis for further discussion, followed by the production of a final polished version;
- with an advanced group, it may be possible to look at the construction of questionnaires from the point of view of manipulation. For example, the following questions, though essentially the same, would probably produce vastly different responses:

1. An old lady in your street has just been murdered for the sake of £2.50 stolen from her purse. Do you believe in capital punishment?

(a) yes (b) no

2. A man convicted of murder three years ago has been set free after another man admitted that he committed the crime. Do you believe in capital punishment?

(a) yes (b) no

- if there is time, this task could be used as part of an integrated activity in which the questionnaire could be used on a target audience, and the results entered into a database and analysed. The sequence of events might be arranged on these lines:
 - deciding on a survey topic;
 - deciding how to obtain a survey sample;
 - formulating questions and making a "dry run" to test the questionnaire;
 - discussing the "fields" or types of information which will be stored on the database;
 - finalising the questionnaire, and wordprocessing it;
 - duplicating the questionnaire;

- carrying out the survey;
- creating the database fields, and entering the results from the questionnaire;
- questioning the database to obtain the required information;
- wordprocessing the results of the survey in various ways – maybe as “newspaper articles” or “company reports” – and incorporating graphs and charts of the results;
- following up with discussion on the findings.

To prepare:

- a worksheet on hanging indents, and a reminder about how to set tab stops, if necessary;
- sample questionnaires from magazines, etc.

NOTES

The integrated activity makes some use of the wordprocessor and database, but a considerable amount of the work is language work, pure and simple. Nevertheless, it is doubtful whether such an extensive project would be undertaken without access to a computer. With the use of the wordprocessor and database, the work can be done professionally and very quickly, and a suitable choice of theme for the survey makes the whole exercise very productive in terms of both language skills and information technology skills.

TASK 18

centring,
underlining,
italics, etc

invitations – cards and letters
letters of reply to invitations

This exercise can include whatever facilities your wordprocessor (and printer) provide for varying text and making it look attractive.

Discuss with students the kind of information which should be included in an invitation, and the different kinds of invitation people send (formal, informal, wedding, birthday party, etc). Discuss the ways in which the presentation of an invitation-card may be made more attractive, stressing the features which are available on your own system. The discussion could include:

centring text

underlining

Varied Use Of Capitals And Lower-case

use of italics or other fonts available

use of **expanded** or condensed characters, if available on your printer

[illegible]

decoration, using the "fancy" characters on the keyboard

Provide printed samples of formal and informal invitation-cards using these techniques, and parallel letters containing the same or similar information, then allow time for the students to construct and print out their own invitation-cards (possibly using data supplied by you).

This method of practising techniques of "pretty printing" is fun, and makes learning about the printer slightly more interesting than simply formatting a prepared text. How difficult it is to do depends on the combination of wordprocessor and printer you are using. Naturally, the task will be easier if you are using a WYSIWYG wordprocessor. If your wordprocessor uses embedded commands, students should be encouraged to use the "preview" option each time they add a new line, to make sure that the effect is the one they wanted to achieve. If there is no preview option, they should do frequent trial prints. This is particularly important when a combination of features is to be used in one line (centred, bold, underlined italics, for example), as it is very easy to forget to "switch off" a command, which will then carry over to the next line, sometimes with undesirable results!

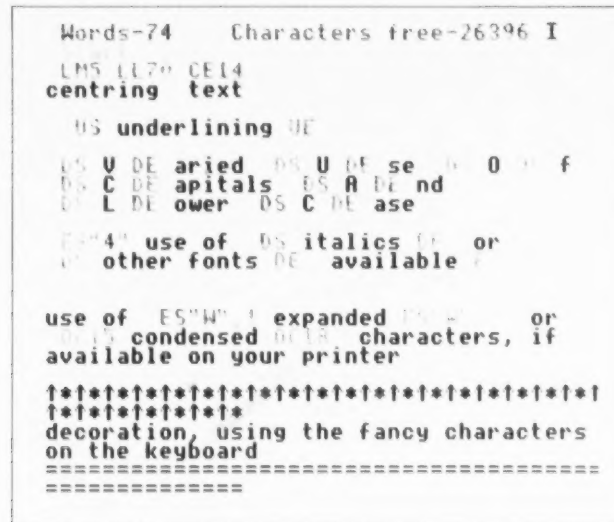


Fig. 6. "Fancy" printing is difficult when you can't see how the finished product will look.
Wordwise Plus (BBC).

Pre-computer work:

- examine the language of invitations, with special attention to the differences between formal and informal styles;
- compare invitations in letter-format and card-format, using the same information if possible;

- if time allows, provide students with information (different for each group) on which to base letters of invitation (perhaps using the wordprocessor to prepare them); use these letters to convert the same information into invitation-cards;
- discuss the form and content of a letter of reply to an invitation – consider why the reply is always in the form of a letter rather than a card – and look at some sample replies.

Follow-up exercise:

- after examining the invitation-cards for style and content, ask the students to swap invitations, and write a suitable letter of reply.

To prepare:

- sample invitation-cards, including perhaps an invitation to a children's party, a wedding, and a party to celebrate a national event/holiday; these could be done on the wordprocessor, but would be more interesting as realia;

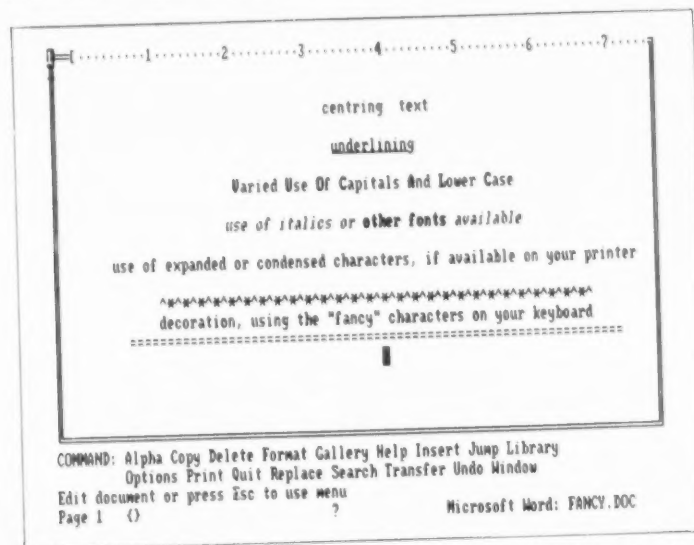


Fig. 7. Word is very nearly WYSIWYG. Word (RM Nimbus)
Figs. 6 and 7 would look more or less the same when printed on paper.

- sample invitation cards prepared on your wordprocessor, making the most of the available printer features;
- sample letters of reply, if possible replying to the sample invitations above;
- a worksheet on how to "switch on" whatever printer effects you want the students to use, *and how to switch them off*.

NOTES:

Attractive formatting of text makes all the difference to the appearance of the finished product, whether it is a letter, an invitation, a list, a report, or whatever.

There are also useful applications of some of the printer features in language work: for example, having learned how to use bold print or underlining, students might be encouraged to look at magazines or newspapers to discover how and when these features are used to draw attention to elements of a text, and the difference between the use of, say, inverted commas and underlining to emphasise a word or phrase. They might then go on to format a sample text (supplied by the tutor), first deciding which elements to emphasise and then applying printer features to them. It is also worth pointing out the risk of over-using these features (they can have the same effect as "crying wolf").

Another facet of formatting which many students enjoy is to type a text in an appropriate shape – for example, a Christmas message in the form of a Christmas tree, or a love letter in the shape of a heart. Remember to turn justification off before attempting this kind of activity.

BEWARE:

Some of the symbols on your keyboard may look different when they are printed on paper!

TASK 19

Wordprocessor task:
block move

Language task:
sequencing

Block move is a function which allows text to be re-ordered on the screen. It is very useful when composing at the keyboard, as the text can be composed in any order, and then re-assembled before printing. This way it is possible to, say, transpose two paragraphs without any re-typing.

Prepare a wordprocessor file containing a readily-identifiable sequence of events, presented in the wrong order – for example, a daily timetable in which each event is preceded by the time at which it occurs, a sequence of instructions for making a pot of tea, or a simplified version of a well-known story. (Hint: if the block move command is new to you, practise by preparing your text in the correct order, **saving the file in case anything goes wrong**, using block move to make a jumbled version, and re-saving the file. Preserve both versions).

To simplify things for students, choose a sequence which consists either of whole sentences "marked" by full stops, or alternatively of single lines of text. Your worksheet should deal with moving just one type of text – either sentences or single lines, whichever is easier on your wordprocessor. Do not confuse the issue by including more than one type of movement at this stage.

Even for beginner readers, it is better to use simple sentences or lines of text for this first exercise, rather than individual words, because it is quite difficult to position the cursor correctly so as to ensure that you move the spaces between words as well as the words themselves.

Sequencing exercises are useful in language work, and the advantage of doing them on the wordprocessor is that you can construct your own texts to suit your students. This activity is therefore one which you can repeat using different texts, and which will still remain interesting. However, you should read the *NOTES* below before deciding how much of this kind of exercise you will want to do on your wordprocessor.

Pre-computer work:

- some students may like to read a printed version of the text (in its correct order) before attempting the exercise;
- it would probably be advisable to work through the first text as a group, deciding on the correct order in advance, so that when students go to the computer they will be able to concentrate on the mechanics of moving the text, more than on applying their language skills to sort out the correct order.

Follow-up exercises:

- more advanced students may want to progress to a more complex exercise where the sequence of the text is less obvious, and may depend on connectors such as "however", "although" and so on, which are much more demanding in language terms;
- once block-moving of sentences and lines has been assimilated, students can proceed to other kinds of block move: paragraphs, phrases and individual words.

To prepare:

- a wordprocessor file containing a text of jumbled sentences or lines;
- a worksheet on how to "mark" sentences or lines (whichever you have chosen), how to move them after marking, and how to close up the gaps left by moving text, if that is necessary on your wordprocessor.

NOTES:

The value of the block move command is only really appreciated when a keyboard composing job is to be done.

It is possible to start a piece of work by jotting down notes on screen in the order in which they occur to you, then re-ordering the notes using block move, and filling them out into a complete text, with no need for scraps of paper covered with odd ideas linked by arrows, and no re-typing.

On reviewing a piece of work, you will often find that although the text is substantially correct, it would benefit from being presented in a different order. This is easily achieved using block move.

And even if students never need this degree of sophistication in their writing, sequencing exercises are useful for improving language skills, and are fun to do. Sequencing on the wordprocessor can of course be integrated with any other activities which you normally do involving the ordering of text.

It must be pointed out, however, that some wordprocessors make block moving much easier than others. If you have the kind of wordprocessor that requires you to mark the beginning of the block, reposition the cursor, mark the end of the block, then move it (**Wordstar** and **Wordwise** do this, as do many other popular wordprocessors), the process is rather tiresome, and once you have learned the technique, you will probably want to use it only when necessary while composing at the keyboard. If you want to do sequencing exercises for their own sake, you can buy programs which will accomplish the same thing (for example, **Word Sequencing** and **Sentence Sequencing**, both published by Acornsoft), although you will probably have to put up with the choice of texts supplied with the program.

If you have a wordprocessor (like **Word**) which allows you to mark the block with a single keystroke, sequencing exercises can be fun to do. And you needn't go to the expense of buying another program. The advantage of a dedicated sequencing program is that it will tell students when the sequence is correct; the advantage of the wordprocessor version is that you can print out a copy of the text (either before or after sequencing, or both) to take away.

TASK 20

Wordprocessor task:
block move

Language task:
paraphrasing/register

Once students have become accustomed to block-moving text, they can begin to use it in more sophisticated ways. This exercise involves re-working a text so that, while its meaning remains the same, the vocabulary, style or register will be different.

Prepare a wordprocessor file containing the text that is to be used, and make a list at the bottom of the text of the alternative words and phrases which you wish to be substituted for the originals in the text. (To make the exercise a little easier, parts of the text which are to be deleted can be marked in some way, either by underlining or – if your wordprocessor is not WYSIWYG – printing in block capitals).

The task is to

- identify a phrase in the text which has a partner in the list;
- delete the phrase from the text,
- move to the list and "mark" the counterpart phrase;
- use block move to deposit the new phrase in the appropriate position in the text.

The language element of this activity can be as easy or as difficult as the students can cope with. It can simply be a question of substituting synonyms. It can be substituting phrases for single words, or vice versa. It can be turning an informal text into a more formal version. It can be changing a layman's explanation of a process into a professional account, by substituting technical language. Whatever the choice, it is wise to double-check carefully to ensure (at least at the first attempt) that substitutions can be made without any other alterations in structure being necessary. This is probably the most difficult part of preparing the text.

Pre-computer work:

- if appropriate to the text you are using, there could be a discussion of register, asking students to identify cases in which the language used might need to be different depending on the situation – the difference between the language of a letter, for example, and the language of conversation, or the language used when among friends compared with saying the same thing at a job interview;
- students could be asked to match vocabulary from one list with its counterparts in another, commenting on the situations in which this vocabulary would be suitable;
- two versions of a story using different styles or registers could be compared;
- students should prepare the language element of the wordprocessor task carefully in advance by discussing a printout of the text they are going to work on.

Follow-up exercises:

- the students will need to check the printouts of their work carefully to make sure that all the substitutions are correct, especially if the text is a tricky one;
- they may like to work through a short printed text, seeing how many substitutions they can find from their own resources, or perhaps using a dictionary or thesaurus; if they do this initially in pairs, then pool their ideas, the number of valid substitutions can be quite staggering;
- it is interesting to examine two versions of the same newspaper story taken from different sources (perhaps from the *Daily Mirror* and *The Times*), and to compare the different effects, especially from the point of view of slang, emotive language, or implicit attitudes.

To prepare:

- a wordprocessor file consisting of a text with a list of alternative words/phrases appended;
- a printout of the file for prior discussion;
- a worksheet explaining the block move procedure, as in the previous task, but dealing with the movement of words and phrases;
- if required, supplementary materials for promoting discussion of style and register, as outlined above.

NOTES

It is quite possible to do this kind of work without using the wordprocessor at all. But apart from the practice in wordprocessing, and the motivation which using the wordprocessor usually adds to any activity, there is the added pleasure of being able to print out a copy of the final version of the work to check for accuracy and to take away afterwards. The value of a printout cannot be overestimated. it is so much better to be able to take away a perfect copy of both versions of the text for future reference, rather than a single copy marked with pencilled notes and arrows, which is neither one thing nor the other!

TASK 21

Wordprocessor task:
search and replace

Language tasks:
information gap
reading comprehension

The "search and replace" function allows you to work through a piece of text consistently substituting phrase B for phrase A. Suppose that you realise that you have wrongly spelt the name "McDonald" as "MacDonald" all the way through a text: search and replace will search for every example of "MacDonald" and replace it with "McDonald" wherever it occurs. On another occasion you may find that you have over-used the word "house", and would like to substitute "home" in some cases: search and replace will stop and check which examples you wish to retain, and which are to be changed.

This task asks students to identify which words are represented by symbols in the text, and to swap the correct word for the symbols using the search and replace function.

Prepare a file consisting of a text in which some of the words are repeated several times. The words should be ones which the students have already met, and which appear *in exactly the same form*¹ in several places in the text. Lexical items may be used, or prepositions, articles and so on.

You should then choose keyboard symbols which are not normally used (\$,] and < , for example) and use the "search and replace" function to work through the text on the wordprocessor, replacing each of your chosen words with one of the symbols.

In the following text, the indefinite article has been replaced with "\$\$\$", and the definite article has become "%%%".

1. If the word is "regular", then forms with and without inflections may be picked up by search and replace, provided that the uninflected form is used as a search criterion - eg. "walk" and "walked" are both permissible, as are "house" and "houses". For other problems which may arise when using search and replace, see the *NOTES* at the end of this section.

At some time in your life, applying for \$\$\$ job by post could be %%% most important letter you ever write.

\$\$\$ letter of application will therefore need all %%% care and attention you can give it.

\$\$\$ letter is %%% first step in getting an interview. It must create \$\$\$ good impression.

The replacements have been made with groups of three symbols because these are easier to pick out visually than single symbols.

Choose with care the words you want to replace, and also take care which options you specify during the search and replace operation. If you have not used search and replace before, read the *NOTES* at the end of this section before starting work.

After making your replacements with symbols, save the file, using the name which you will require your students to use, and then make a copy of the file using a different name. Use this second copy (in conjunction with the worksheet which you have prepared) to go through the procedure which you will be asking your students to use – that is, replace all your symbols with words again – to check that the text and replacements are within the scope of the students, but also to check that there are no hidden difficulties in the task. Keep a note of the options you have specified in making the replacements, so that your students will be able to specify the same options when they do the exercise.

Pre-computer work:

- if there is a specific structure element in your text, such as definite/indefinite articles or prepositions, the usual preparation will need to be done in advance of this task;
- if the text involves lexical items, some preparation may need to be done on introducing any new lexis;
- it may be a good idea for students to read a complete version of the text before starting work at the wordprocessor.

Follow-up exercise:

- in addition to checking the printouts for accuracy, it would probably be advisable to provide some reading comprehension work based on the text. This is because, having

found the correct word to replace a symbol, the replacements can be made throughout the text without any necessity for reading through the whole piece: students need read only as far as necessary for them to identify the meaning of the symbols. Reading comprehension work would therefore ensure that they make the most of the text.

To prepare:

- a file consisting of a text containing suitable vocabulary replaced by symbols, thoroughly checked;
- a worksheet explaining as simply as possible the "search and replace" procedure, and how to set the options required for the exercise (the sheet should also explain how to put the procedure in reverse, so that students can rectify any wrong replacements they may make);
- a handout containing a version of the text in its complete form, for students to check their work;
- a reading comprehension exercise based on the text.

NOTES:

Of all the tasks to date, this one probably has the most potential pitfalls, because of the idiosyncrasies of the search and replace procedures of individual wordprocessors. It would be advisable for you to familiarise yourself thoroughly with the operation on your own wordprocessor, before inflicting this task on your students!

Here are some of the points to watch out for (make allowances for the fact that the terminology used for your wordprocessor may be slightly different):

- *there is usually an option which allows you to choose between "global" and "selective" replacement: it is usually better to opt for "selective" replacement, which means that the cursor will stop at every example of the text you are searching for, and allow you to decide whether to replace it or not; this has two advantages, in that it allows the replacement to be checked, so that if a mistake has been made it can be rectified before too much damage has been done, and it forces students to read more of the text than they otherwise might have done. "Global" replacement replaces every example in the text without any check;*

- there is often a choice of "whole word search" which means that a search for the indefinite article "a" will ignore the letter "a" in the words "apply" and "create"; it also means, however, that the search won't find the "a" of "an"! In the same way, a whole word search for "letter" will not find "letters"; you will have to decide whether you want "whole word search" or not, and make sure that the students make the same kind of search as you did when you created the text;
- some wordprocessors allow you to "ignore case", which means that a search for "the" will also find "The" at the beginning of a sentence; beware of using this lightly, however, because the replacement will be made using exactly what you have specified – so that a word at the beginning of a sentence will be replaced without a capital letter; there are wordprocessors which do not allow you to "ignore case", and here the search will find only exact matches – if you specify "the", the program will not understand that "The" is a version of the same word, and will ignore it; if you construct your text with care, this kind of problem can be avoided;
- some wordprocessors have "dynamic formatting" – that is, if you replace a word with another word of a different length, the lines of text will be re-organised on the screen automatically; others, like **Wordstar**, will leave lines of odd lengths after the search and replace procedure, and you will need to explain in your worksheet how to re-format the text so that it will look right on the screen and print out properly;
- most wordprocessors will search either forwards or backwards in the text, so that if your cursor is at the beginning you search forwards, and if it is at the end you search backwards; many search and replace facilities, however, will only search from the cursor position, so if your cursor is in the middle of the text, the search will be made from that point, and will not cover the complete text; if your wordprocessor is like this, the first instruction on your sheet should be "Move the cursor to the beginning of the text", and then all searches can be made forwards, covering the whole of the text;
- some wordprocessors only have extremely rudimentary search and replace facilities which do not allow you to specify useful options, and you may find that the only educational value to the students is learning how to do the search and replace function itself, because it is impossible to use the facility for any valid language activity; in this

case, you have to decide whether you want your students to learn the function for its own sake, or whether to ignore it completely.

You would be well advised to save a version of the complete text before starting to experiment with search and replace! It is always useful to keep both versions for future reference, in any case.

In wordprocessing generally, search and replace can be invaluable. It can quickly make changes to a text which would take a considerable time if done by hand. It can help you to correct spelling mistakes, to replace names or places with specified alternatives, or to mark a point in the text (with a little-used symbol such as @) so that you can easily return to it at a later stage. It can even save you typing-time. Suppose you work at the High North Road and Ealingbury Adult Education Institute, and are typing a text in which the name of the institute appears many times. Simply use an abbreviation (in this case, say "hn") every time you want to use the name, then when the text is finished, search for all the examples of "hn" and replace them with "High North Road and Ealingbury Adult Education Institute".

BEWARE:

use a selective search at all times, even when you think you are sure that every example needs to be replaced! Global searches can produce very odd and unexpected results. Suppose you want to replace the word "tab" with "tabulate" every time it occurs. You choose global search, and forget to specify "whole word only". What happens to "suitable" and "database"?

TASK 22

Wordprocessor task:
search and replace

Language task:
homophones

This task is a variant of the previous one, but here the students are asked to fill the asterisked gaps with one of a list of known words, using search and replace.

Many students have difficulty in distinguishing between the spellings and meanings of similar-looking words. The sample exercise below takes three commonly confused forms (*there*, *their* and *they're*) and asks students to choose the correct one to substitute in each gap.

Doc: edphone Mode: Revise
Page=1 Line=18 Col=6
-----L-----R-----

Many secretaries learn * skills in colleges, where * involved in courses of various kinds.

While * at college, they hope to learn shorthand and typing, and perhaps improve * spelling while * *, too.

Nowadays, * is increasing pressure to learn wordprocessing and other computer skills, and secretaries find that * employers assume that * familiar with a wide range of information technology applications.

In the present economic climate, * is a real incentive to be "computer literate", because employers are increasingly computerising * businesses and so * always on the lookout for staff with suitable experience, which can therefore be an important asset to secretarial staff when * jobhunting.

Save text & return to menu (y/n) ?

Fig. 8 Edword (BBC).

The task is for students to use search and replace three times to complete the text. On the first run, they will search for * and selectively replace it with *there*, reading through the text and making a decision each time the cursor stops at an asterisk about whether or not *there* is a suitable replacement in that position. The cursor is then returned to the top of the text, and another selective search and replace operation is carried out to replace the asterisks with *their*. The final pass picks up the remaining asterisks and checks to see that they all represent *they're*.

Pre-computer work:

- introductory work on the chosen homophones;
- reading the passage thoroughly for meaning before embarking on the wordprocessing part of the activity.

Follow-up exercises:

- a cloze exercise as in Task 11, using the "insert mode method", and clozing some of the homophones practised in this task;
- free composition, making use of the homophones used in the exercise.

To prepare:

- a text in which the homophones to be practised are replaced by symbols;
- a worksheet covering the search and replace procedure, reminding students that they should use a *selective* search, and that they should return to the start of the text before beginning each new search;
- possibly a supplementary cloze text practising the same homophones.

NOTES:

Conventional multiple-choice exercises on homophones are not the most exciting way of practising language. This activity gives the opportunity to achieve the same object in a more interesting way, and can additionally provide the student with a printed copy to take away if required.

TASK 23

Wordprocessor task:
search and replace

Language task:
vocabulary building/idioms

Prepare a file consisting of a text in which the same word is used several times, but with varying meanings.

The verb "throw", for example, might be used in "throw a ball", "throw a party", "throw cold water on something" or "throw on some clothes", where "throw" would mean in turn "toss", "give", "pour" or "put". Another example might be various uses of the word "nice", as in "a nice day", "a nice meal", "a nice surprise", "a nice person".

The expressions should, if possible, be combined into a continuous text, if this can be done without stretching credibility too far; otherwise incorporate them into separate sentences.

Write your text in such a way that

- some of the interpretations of your chosen word appear more than once;
- each example of the word can be replaced by one (or more) words *without any further structural or lexical changes being necessary*;
- the target word appears *in the same form* throughout the text, otherwise there will be difficulties with the search procedure – for example, if the verb "throw" appears in its present and past forms in the same text.

The following text uses the verb "throw" in a number of different ways.

Ronald was going to be seventeen soon. He had never had a birthday party before, and he wanted to throw a party in his own honour. When he mentioned the idea at home, however, to his surprise his family started to throw cold water on it. He was so hurt that he decided to throw on his anorak and go off to find his friends. He explained to them that he wanted to throw a party, and was even more hurt when

they began to throw cold water on the idea, too. Unhappily, he looked around for a ball to throw against a wall – anything to work off his anger! – and after a few minutes of bad-tempered exercise, he cooled off and thought it would be better to go home and throw himself into his school work to take his mind off the party he wasn't going to have.

When his birthday arrived, he spent the day at school as usual. Nobody wished him a happy birthday. When he got home the house was empty, and he went miserably upstairs to throw himself on his bed, where he tried to occupy his mind with a book. After an hour or so, his brother knocked on his door: "Come on, hurry up, throw on your coat, I've got a surprise for you!" Puzzled, Ronald hurried to throw on his outdoor things, and went with his brother to his uncle's house, where all his family and friends were waiting for him. They had been secretly planning to throw a surprise party for him after all!

The students' job is to replace each example of the chosen word with a suitable synonym, using the "search and replace" function. In a text exploiting the word "nice", if the first occurrence happened to be "a nice day", the students might decide that an appropriate word would be "sunny" so they would call up "search and replace", asking the program to search out examples of "nice", and replace with "sunny". There are two ways of doing this.

1. They can use the option "check before replacing" (the opposite of "replace without asking"), so that the cursor stops at each example of "nice", and they are asked to confirm whether they wish the replacement to be made. This should lead to discussion of the vocabulary involved, and the (un)suitability of the various replacements in each occurrence of the chosen word.
2. They can use a "global replace" so that every example is replaced automatically (so they would ask for every occurrence of "nice" to be replaced by "sunny"). They then return the cursor to the start of the text and do a *selective* replace in the opposite direction (asking for examples of "sunny" to be changed to "nice"). The cursor will now stop at every occurrence of the word "sunny" so that the students can examine it to see whether it is a viable alternative to "nice" in that context. If it is, it stays; otherwise, it is changed back to "nice". They then repeat the process until all possible changes have been made.

Both methods lead to the same result, but the second produces considerable amusement as students see the absurdity of some of the changes on the screen – for example, "we had a sunny

meal". Other changes can produce interesting arguments – for example, is it possible to talk about "a sunny person"? After all, you could refer to "a person with a sunny disposition".

You may like to include in the text some expressions for which no replacement is possible – for example, the word "throw" in "to throw yourself into your work" can not readily be replaced with another single word. (A near equivalent like "apply" involves another change, in that you apply yourself *to* something, not *into* it).

After working on the sample text about Ronald's birthday, the resultant story might read like this:

Ronald was going to be seventeen soon. He had never had a birthday party before, and he wanted to *give* a party in his own honour. When he mentioned the idea at home, however, to his surprise his family started to *pour* cold water on it. He was so hurt that he decided to *put* on his anorak and go off to find his friends. He explained to them that he wanted to *give* a party, and was even more hurt when they began to *pour* cold water on the idea, too. Unhappily, he looked around for a ball to *toss* against a wall – anything to work off his anger! – and after a few minutes of bad-tempered exercise, he cooled off and thought it would be better to go home and *throw* himself into his school work to take his mind off the party that he wasn't going to have.

When his birthday arrived, he spent the day at school as usual. Nobody wished him a happy birthday. When he got home the house was empty, and he went miserably upstairs to *throw* himself on his bed, and tried to occupy his mind with a book. After an hour or so, his brother knocked on his door: "Come on, hurry up, *put* on your coat, I've got a surprise for you!" Puzzled, Ronald hurried to *put* on his outdoor things, and went with his brother to his uncle's house, where all his family and friends were waiting for him. They had been secretly planning to *give* a surprise party for him after all!

**Pre-computer
work:**

- the vocabulary should be introduced in advance, so that its meaning is thoroughly understood, and alternative ways of expressing the same idea are elicited: discuss as many different ways as possible of achieving the same meaning for an idiom, and be sure to

include (wherever possible) at least one which involves *changing only the target word*, since this is the exercise that the students will be doing at the computer.

Follow-up exercises:

- discuss the students' printouts, checking that the correct part of speech has been substituted for the target word, and that the new expression makes sense without any further alteration to the target phrase;
- give the students a suitable text and ask them to substitute the idioms they have been working on in the appropriate places: if this is done as a pencil-and-paper exercise, more extensive changes can be made, involving a whole phrase rather than just the target word, and the target word can appear in any form.

To prepare:

- a wordprocessor file containing suitable vocabulary;
- a worksheet explaining the "search and replace" procedure, and how to set the options required for the exercise;
- supplementary practice materials as described above.

NOTES:

For detailed notes on search and replace, see Task 21.

On some simple wordprocessors, the search and replace facility may not allow you to check each occurrence before replacing, and this kind of limited facility is obviously unsuitable for this exercise.

TASK 24

Wordprocessor task:
search

Language task:
roots and suffixes

The search function allows you to specify a word or phrase to pinpoint in a text, and the cursor will stop at each example of that word or phrase in turn, so that you can change it or simply look at it in context. This can be a useful way of encouraging students to examine the language more closely.

There are two possible ways of tackling this exercise:

WORD-ENDINGS Prepare a wordprocessor file containing a number of different examples of words using each of the word-endings you want to work on – for example, –ed, –ly, –tion, –ful, –ing. They should be in context, in discrete sentences if need be, but if possible as part of a continuous text. Ask students to use the search facility to track down each of the examples in turn, and identify what part it plays in the sentence (This would entail first searching for all the examples of –ed, then all the examples of –ly, and so on). The exercise can be extended, if required, by asking students to jot down on paper (or list at the foot of the screen) other examples of the same part of speech which could be substituted in that context (not necessarily with the same meaning, but still keeping a meaningful sentence). If you feel the students can cope, you could include one or two non-standard examples – perhaps “wicked” (which is not the past form of a verb) or “smelly” (which is not an adverb) – and get the students to pick them out.

Prepare a different text using the same endings, but this time omit the roots. Students can then search for the endings, and find suitable roots to complete the text in a meaningful way.

ROOTS

Prepare a wordprocessor file containing a text which uses one word-root (or more, if the students can cope) used in a variety of different parts of speech – for example:

obey, obedience, obediently, obedient
false, falsely, falsify, falsehood
true, truly, truth, truthful

Ask students to search for the root, and find and identify all the different parts of speech in which it is used. Make sure that you supply the root in its most useful form: the root for the word "false", for example, is "fals" – otherwise, the search will not find "falsify".

Prepare another file using the same roots, but this time omit the endings. Students can then search for the roots, and type in the endings to complete the correct part of speech.

Pre-computer work:

- preparatory work on parts of speech, especially the use of suffixes, with attention to consequent spelling changes;
- preparation of a "dictionary" (stored on disc?) of word-families, which can be extended from time to time as new families are found;
- prepare the wordprocessor texts in advance, using them to generate as much vocabulary as possible before going to the computer.

Follow-up exercise:

- continuing work on the word-families dictionary.

To prepare:

- a worksheet explaining the "search" procedure, and how to set the options required for the exercise (if any);
- a wordprocessor file containing a text using word-endings or roots for the search-process;
- a second file to be completed by the students with word-endings or roots as appropriate;
- blank "dictionary" sheets, either as wordprocessor files to be completed in overtype mode and printed out, or as printouts to be completed by hand. They could look something like this:

ROOT	NOUN	VERB	ADVERB	ADJECTIVE
fals-	falsehood	falsify	falsely	false
etc.				

NOTES:

Many of the same criteria apply to "search" as to "search and replace" (see the NOTES for Task 21 where search and replace is discussed in more detail).

If your wordprocessor provides an option which allows you to search for "whole words only", it is best to say "no" to this (in this exercise), as by definition the roots and suffixes for which you are searching will be parts of words, and not whole words.

Be sure to stress in your worksheet the importance of returning the cursor to the beginning of the text before starting a new search, so that the search will cover the whole text and not just forward from the cursor position.

Search is a useful wordprocessor facility, as it allows you to move quickly to a portion of text that you wish to examine. For example, suppose that in a long text you wish to refer to specific page numbers, but you are not sure at the time of typing what these numbers will be: typing "page X" on each occasion means that later you can search through the text for each occurrence of "X" and change it for whatever number is needed at that point. Similarly, you could mark, say, a point in the text where you need to insert information which is not immediately available to you. (Some wordprocessors, like **Wordstar**, have a separate marker facility just for this purpose).

Search should be used with care, because it is easy to bypass what you are looking for with a careless search. An example of this occurred with the root "fals" described above – it would be easy to miss some relevant material by making too specific a search. On the other hand, searches can come up with some surprising matches if you are not specific enough: a search for the indefinite article "an" will come up with "language", "bank" etc., unless you specify a "whole word" search.

The trick with search is to specify the optimum amount of text which will find what you want as quickly as possible – and this is a skill in itself.

TASK 25

Wordprocessor tasks:
block copy
windows

Language task:
summary writing/précis

The "block copy" facility on the wordprocessor is similar to "block move" (which was used in the sequencing exercise in Task 19), except that the marked block¹ is not erased from its original position, only copied to the new position.

A "windowing" feature will only be found on more sophisticated wordprocessors (such as *Word* on the Nimbus). The use of screen windows allows two (or more) files to be edited at the same time, appearing in different screen areas or "windows". *Wordwise* + users may wish to try using "segments" as an alternative to windows. Segments allow you to work on two files, although they may not both be present on the screen at the same time. If you have neither windows nor segments, the task can still be done, though less conveniently.

Prepare a wordprocessor file containing the text which is to be summarised. Discuss a printout of this thoroughly with the students, deciding with them which parts of the text are essential to the summary, and which may be omitted. The activity will be done using one of the following methods, depending on what kind of wordprocessor you are using.

Method 1: Windows

- divide the screen into two horizontal windows, and load the original text into one of them (preferably the top one);
- mark the first block which is essential to the summary, and copy it to the second window;
- mark the next block and copy it into the second window below the first block;
- continue to do this until all the essential elements of the text have been copied to the second window;

1. Although these notes refer to "blocks" of text, point out to students that a "block" for these purposes can be anything from a word to a paragraph, stressing that the essence of summarising is brevity.

- clear the original text, leaving the top window empty;
- the bottom window will now contain an abbreviated version of the text, though still using the same words as the original;
- write the summary in the top window, using the bottom window for reference: because you are using windows, you will be able to refer to any part of the text in the bottom window without disturbing the summary you are writing in the top window;
- once all the ideas from the bottom window have been included in the summary, close the bottom window, and use the full screen for the "polishing" process to finish the summary;
- save and/or print the summary.

Method 2:

Segments

(for *Wordwise* +)

- load the original text to the screen;
- repeat the following procedure as many times as necessary to copy all the essential blocks into a segment:
 - mark the block you want to copy;
 - press ESCAPE to go to the main menu;
 - use menu option 3 to "save marked text", giving the text the name TEMP (you will need a disc in the drive when you do this, of course);
 - use menu option 9 to go to the segment menu;
 - use menu option 4 to "load text to cursor", and give the filename TEMP when you are asked for it;
 - press ESCAPE to enter the segment, then press RETURN to move the cursor to a new line, then ESCAPE again to go back to the segment menu;
 - use menu option 9 to go to the main menu;
 - press ESCAPE to get the original text on the screen;
- note that however many times you copy a block in the above procedure, you can use the same filename TEMP;

- when you finish copying blocks, go back into the segment, which now contains an abbreviated version of the original text, though using the same words;
- type the summary below the existing text, deleting the existing text as you use the ideas from it in the summary (you can use "block delete" for this);
- save and/or print the segment, once the summary is complete.

Although this sounds rather complicated, it can soon become a matter of routine. It is less convenient than windows, because you can not have the original text on the screen at the same time as you build the abbreviated version, but nevertheless it works.

Method 3: Single screen

- mark the first block you want to copy;
- move the cursor to the bottom of the text;
- copy the block into its new position;
- repeat this procedure until you have copied all the necessary blocks;
- use "block delete" to erase the original text from the screen;
- you will now be left with an abbreviated form of the text, but using the same words as the original;
- type the summary below the existing text, deleting the existing text as you use the ideas from it in the summary (you can use "block delete" for this);
- use the full screen to "polish" the final version;
- save and/or print the summary.

Although the instructions for this method look much simpler than for windows or segments, remember that students will need to do a considerable amount of moving backwards and forwards through the text using the single screen method, and this can be confusing. Both windows and segments avoid this problem to a large extent, and with windows there is the additional advantage that you can see both old and new versions of the text on the screen at the same time, but in quite distinct areas.

Make sure that, whichever method you are using, students are fully conversant *before they start work* with the techniques they will need in order to copy the text and write the summary.

Pre-computer work:

- introduce the idea of summarising, using realia if possible: newspaper articles can be compared with "stop press" paragraphs on the same subject, if you have access to a video, compare news headlines with the full news report, and if you have teletext, use the news summary pages in comparison with the expanded story pages;
- provide several pieces of text for practice, asking students to discuss which parts ought to be retained in the summary, and highlight them using marker pens;
- provide sample sentences for students to paraphrase, asking them to retain the original sense, but to use different – and, if possible, fewer – words.

Follow-up exercise:

- provide suitable texts for students to prepare individually, asking them to mark the essential points, or make notes of them, then to summarise the text orally, within a given number of sentences.

To prepare:

- a worksheet containing thorough notes on the procedures needed for the task – including block delete if it is to be used – and also a summary of the instructions, for use as the procedure becomes more familiar (this could actually be used as realia to demonstrate the technique of summarising!);
- a wordprocessor file containing the text to be summarised;
- materials for demonstrating and practising summary techniques, as described above.

NOTES:

The procedures for summarising described in this task are in some ways rather cumbersome. But apart from practice in the new wordprocessing techniques involved, this kind of exercise has another value: it directs students' attention to vital parts of the text, and provides an additional step in the process of refinement, which many students find necessary when they first start summarising.

The problem with pencil-and-paper summarising is that the job is so laborious that few people have the patience to break it down into stages – first identifying the essential elements, then isolating them from the original text, and finally paraphrasing them to form a summary. The skill of isolating the essentials directly from the original comes with practice: for most students, the extra stage in the refinement process is vital, and the computer can help with this.

Most students may never need to use summarising techniques; but many will need to acquire note-taking skills, which are a further refinement of summarising – that is, isolating the essentials from a verbal original. Summarising can be a useful intermediate step in learning successful note-taking strategies.

The block copy facility is a useful one to learn, and very simple once block move has been mastered. One instance where block copy is very useful is in building up a chart, where there is a lot of repetition. After typing one section correctly, this can then be copied as many times as necessary to form a complete chart. So this:

.....	
-------	-------

can very quickly become this:

.....	
.....	
.....	
.....	
.....	

by using block copy. The chart can then be printed out blank, or can be filled in on the wordprocessor, using otype mode.

The windows facility is invaluable to anyone who uses the wordprocessor regularly. It is common to find that in the middle of working on one text, you need to refer to some information in a different text: simply make another window on the screen, and load the required text into it. And of course, if you want to transfer some of the information from one window to another, it is simple to block copy it. When you have finished with the second window, it can be closed and the full screen is available once more.

TASK 26

Wordprocessor task:
merging files

Language task:
group composition

The "file-merge" facility allows work done by different people, or on different occasions, or different machines, to be brought together as a single piece of text.

You may find that your wordprocessor uses a different name to describe file-merging – *Wordwise*, for example, calls it "loading text to the cursor" (option 4 on the menu), and this is a very good description of what happens: you position the cursor at the point where you want the new text to appear, and give the command for the new file to be merged into the existing one at that point.

For the group composition activity, select a text (possibly a newspaper story, or an incident from a book), and write the opening words of three paragraphs, representing the beginning, the middle and the end of the chosen story. These paragraph-openers should offer some context from which the body of the story can reasonably be deduced.

The students see all three paragraph-openers, but not the original text. Then, working in three groups, each group attempts to construct one of the three paragraphs of the story, using the wordprocessor. A suitable length for each paragraph should be agreed before work is started.

When all three groups have finished work on their individual paragraphs, these are saved, then copied on to one disc, and the files merged to make a complete story. Each student can then have a printed copy of the story.

An alternative might be to prepare a larger number of paragraph openers in the form of a game of Consequences.

Pre-computer work:

- give the students a few sample paragraph-openers, and ask them to develop these orally into a brief story; point out that in order to link the paragraphs successfully, you must pay attention to the paragraph-openers fore and aft as well as your own;
- give the students a sample story, and ask them to write the paragraph-openers themselves

Follow-up exercises:

- the students' version of the story can be read by the whole group and examined for inconsistencies in the story-line;
- their version could be compared with the original text, to see how much/little of the story has been correctly deduced from the context of the paragraph-openers;
- each group could then return to the wordprocessor to "polish" its own (or another group's) paragraph, before re-merging the files to make a final version to be printed out;
- if appropriate, the students might be told in advance the source of the story, and asked to compose the paragraphs in a suitable style for that source. The original and student versions could then be compared stylistically, as well as for content.

To prepare:

- paragraph-openers for the group composition;
- a worksheet on file-merging;
- supplementary materials as described above.

NOTES:

*File-merging is a useful technique, especially if you need to prepare very long texts and/or you are using a wordprocessor like **Wordwise**, which does not take to long texts very happily. File-merging means that you can prepare your text in conveniently-sized chunks, and only put them together when you are ready to print them.*

Another use for file-merging is for bringing together pieces of text written at different times, perhaps even for different purposes. Perhaps you are writing a report which needs to draw on past correspondence or other wordprocessed documents: instead of typing all the required information again, the relevant files can be merged into the main report, and then amended to suit their new environment.

Even if you think you will not need to do this, it is worth bearing in mind that some of the documents you are typing may be worth keeping for future use, and to this end, you should always keep paper copies of your files, labelled with the disc-name and/or number, and the filename, so that you can readily identify them whenever they are needed in the future. It is easy to store materials on disc, and not expensive as one disc holds a considerable amount of text; it is therefore a good idea to retain on disc anything which you feel may be needed again – it is much easier to amend an old file than to start typing a new one from scratch.

TASK 27

Wordprocessor tasks:
merging files,
glossary,
block move

Language task:
rebuilding a text

Prepare in advance a long text, divided into sections and saved as a number of separate files. The text may be as long as you feel is reasonable for students to take in at a single session. The separately-saved sections can be individual sentences, paragraphs, or even longer pieces, but the text should be broken up in such a way that the students have a reasonable chance of deducing the logical order of the different pieces.

The "given" information at the start of the exercise is simply a list of the names of the various files which must be merged to make the complete text. The list should be deliberately mixed, and the names of the files should give no indication of the correct order of assembly!

The students' job is to load a file, then merge with it a second file selected from the list. Having read the files, the group decides jointly on the order in which these pieces of text would probably appear in the final story, reversing them (using the block move facility) if necessary.

Another file from the list is then merged with the first two, a decision made about its probable place in the text, and the contents of that file moved into position. And so on, until all the component files have been merged into one long text.

Each group then prints out a copy of its version of the finished product.

Glossary:

An alternative version of this may be prepared using the "glossary", if your wordprocessor has this facility. The glossary allows you to store useful individual pieces of text in a single glossary file, so that they can easily be called into any text you are working on.

Using *Word* on the Nimbus, for example, you would type your text as normal, then mark a section of it, and copy the marked section to the glossary, giving it a two-character name. The remaining sections would be marked in the same way and copied to the glossary, each piece having its own

name. On completion, the glossary is saved, and is then available for use at any time in the future. It would be convenient to make a special glossary to hold the contents of this task, so that it could be kept and re-used as required.

For the purpose of this exercise, the students would simply be given a list of the glossary names which are needed to build the text. They would type one of the names, then press the F3 function key on the Nimbus, and the appropriate section of text would appear on the screen. The task then proceeds as for file-merging, but using the glossary instead of separate files.

Pre-computer work:

- this task is similar in concept to the sequencing task associated with the block move facility in Task 19, except that here the sections of the story are added one by one, which demands that students use only those clues which are available from the visible text; it might be useful to do a similar exercise using a paper text cut into pieces, giving out the pieces one by one at random, and asking students to talk through the process of assembling them into the correct order

Follow-up exercises:

- the groups jointly discuss their versions of the text, and argue the pros and cons of any alternative orderings,
- they could also consider at what point in the reconstruction the text began to make sense to them, which component files caused problems, whether the original placement of each file needed changing in the light of subsequent files, and so on.

To prepare:

- a text broken up and saved as separate files, or alternatively saved as separate glossary items,
- a different text, printed on paper and cut up into pieces,
- a worksheet covering *either* file-merging *or* glossary, depending on which you have decided to use,
- a list of the filenames (or glossary names) which are to be merged;
- a reminder worksheet giving a summary of the block move procedure; it would be a good idea to have this available, even if you think it will not be needed.

NOTES:

If your wordprocessor has a glossary facility, you should quickly be able to find many uses for it.

Any piece of text which you need repeatedly can be stored in the glossary. Suppose you frequently write letters to the same person: that person's name and address can be stored in the glossary and recalled with a couple of keystrokes every time you write him/her a letter. The glossary will store not only the name and address, but the correct letter format as well, with all the line-endings and/or tabs exactly as you saved them originally.

If you regularly use a dotted line to divide sections of text in a file, you can save the dotted line in the glossary, so that you need never type all those dots again. Any name or other piece of text which you commonly need – add it to the glossary.

Because it will recall all the formatting instructions, the line layout etc., along with the text itself, the glossary is especially useful for saving things which are in an unusually complicated format – for example, if you often type the name of your workplace in bold underlined italics, centred on two lines, the glossary will save this format exactly as you type it, and reproduce it on demand.

TASK 28

Wordprocessor tasks:
files merging
file copying

Language task:
writing up a discussion

This task could well be spread over two or even three sessions.

In the first session, the students discuss, as a group, a controversial issue, and uncover a variety of points of view on the subject.

Each group then goes to the wordprocessor, and types up its own point of view, which is saved as a file. This may be a single sentence, or something more substantial, by prior agreement.

The files are then merged into a single text and the result is printed out, with a copy for each student. The merged file will need to be copied on to each student disc. These jobs may be an integral part of the activity, or could be done by the tutor between the second and third sessions.

Each group then goes back to the wordprocessor, to work on the complete text, moving the paragraphs if necessary, removing repetitions or inconsistencies, and finding suitable links to build the text into a coherent argument.

Pre-computer work:

- the discussion which is to provide material for the written work.

Follow-up exercise:

- the students bring printouts of their final version to share with the other groups, and the whole group discusses the means which have been used to link the arguments, and any social/semantic inferences which may be drawn from the way the links have been made. Is it possible, for example, to deduce the opinion of the writer from the choice of link words and phrases?

To prepare:

- a worksheet on file-merging;
- a worksheet explaining how to copy a file from one disc to another, if the students are to do this part of the task themselves.

NOTES:

Until now, all the work done by students has taken place on one student disc, or in the case of earlier file-merging tasks, it has been assumed that the tutor would take on the job of transferring the students' files from disc to disc before merging them.

In this task, the merged file has to be copied back to the students' working discs, and it might be a good opportunity for each group to learn how to do this for themselves.

One of the aspects of computing which newcomers find most bewildering is that of disc-handling – formatting discs (that is, preparing them for use), copying files and discs, deleting unwanted files, and so on. The best way to learn how to do these jobs is to do them when they are needed. That way, it is easy to understand why you are learning a particular procedure, which in turn makes it easier to follow what needs to be done.

TASK 29

Wordprocessor task:
mailmerge

Language task:
style and register

"Mailmerge" is commonly used to "personalise" circular letters, to give them the appearance of being written individually instead of being "mass produced".

This is done by preparing a template in which the body of the letter contains special markers indicating points at which personal information is to be inserted. The template might begin something like this:



Fig. 9. Word (RM Nimbus).

This template is then "merged" with a list of names and addresses, and each time the letter is printed, a different "personal" letter is produced, looking something like this:

Ms. Anita Barker,
17 Larches Grove,
East Westerham,
Oxhamptonshire,
OX5 3EW.

Dear Ms. Barker,

Congratulations! You have won a major prize . . .

Figure 9 was written for the *Word* wordprocessing program on the Nimbus; if you are using a different wordprocessor, the template will use different "signals" at the point where the variable information is to be included, but the end result should be the same.

The mailmerge facility should also make it possible to insert information from the keyboard, so that the bulk of the letter (or other document) is prepared in advance, but certain information is supplied immediately before printing. Suppose, for example, a circular letter goes out regularly to people on a mailing list, and much of the letter is standard, but the date is of course different each time the circular goes out. Instead of putting "date as postmark", or just not dating it at all, the template could contain an instruction causing the program to stop and ask the keyboard operator to insert the date before continuing to print out copies of the letter to everybody on the mailing list. The template would begin like this:

«SET date=?»

«date»

«title» «fname» «sname»,
«add1»,
«add2»,
«add3»,
«pcode».

Before printing the letters, the program would display the message "date?" on the screen, and wait for the typist to enter the current date from the keyboard. On printing, the variable «date» would

be replaced by the date which had just been entered, and this would appear on every letter in the mailing list.

It is this feature which is used in this task. The template will contain variables which the student will enter at the keyboard, and the result will be a document of which the form will be pre-determined by the template, but major features of the content decided by the student. The following template was written for the *Wordstar* wordprocessor:

&add1&
&add2&
&add3&

&date&

&coname&
&add4&
&add5&
&add6&

Dear Sirs,

I am writing to complain about the &product& I bought from &coname& on &date2&.

Although it was all right at first, it now &wrong1& very badly, and it doesn't &wrong2& properly.

As the &product& is still under guarantee, I am writing to ask for an immediate &rep&, and I look forward to receiving this as soon as possible. I am returning the faulty &product& under separate cover.

Yours faithfully,

&name&
&ms&

The students need not see this template at first, though it may be interesting for them to compare it later with the finished product. The "interrogation file" calls for information which will complete the template letter

```
..letprint.txt – prints a letter of complaint
.av "What is your name?, "NAME
.AV "Are you Mr., Mrs., Miss or Ms.,? ",MS
.AV "Your address? (3 lines) Line 1: ",ADD1
.AV                                     Line 2: ",ADD2
.AV                                     Line 3: ",ADD3
.AV "What is the date? ",DATE
.AV "You want to make a complaint about something you
    have bought which doesn't work properly.
    Complete the sentence: I bought a ...",PRODUCT
.AV "When did you buy it? ",DATE2
.AV "Where did you buy it? (name of company) ",CONAME
.AV "Type the company's address (three lines):
    Line 1: ",ADD4
.AV "Line 2: ",ADD5
.AV "Line 3: ",ADD6
.AV "Write a word to complete this sentence:
    'Although it was all right at first, it now ....
    very badly.': ",WRONG1
.AV "Write a word to complete this sentence:
    'It doesn't .... properly.': ",WRONG2
.AV "A refund or a replacement? Complete the
    following sentence: 'I want a ...'",REP
.FI B:COMPLAIN.LET
```

The students see the parts between the double quotation marks. Everything else simply tells the wordprocessor what to do, and although it looks complex, it is in fact very repetitive, so once the formula has been learned, the preparation of such a file is simple.

The questions will appear one at a time, and when completed, what the students see will be something like this:

What is your name? Isabella Santos

Are you Mr., Mrs., Miss or Ms.? Ms.

Your address? (3 lines) Line 1: 25 Trafalgar Street

Line 2: Hackney

Line 3: London, N.16

What is the date? 12th December 1986

You want to make a complaint about something you
have bought which doesn't work properly.

Complete the sentence: I bought a ... wristwatch

When did you buy it? 1st December 1986

Where did you buy it? (name of company) Jones & Co.

Type the company's address (three lines):

Line 1: 12 Disraeli Place

Line 2: London

Line 3: W.3

Write a word to complete this sentence:

'Although it was all right at first, it now ...
very badly.': loses

Write a word to complete this sentence:

'It doesn't properly.': wind

A refund or a replacement? Complete the
following sentence: 'I want a ...' replacement

In some wordprocessors, the template and the interrogation will form part of the same file, but what the students see will be similar to the above example. As soon as the questions have been answered by the students, the completed letter will be printed. The wordprocessor will organise the line-endings correctly, no matter how long or short the students' answers are, and the finished letter will read:

25 Trafalgar Street,
Hackney,
London, N.16.

12th December 1986

Jones & Co.,
12 Disraeli Place,
London, W.3.

Dear Sirs,

I am writing to complain about the wristwatch I bought from Jones & Co. on 1st December 1986.

Although it was all right at first, it now loses very badly, and it doesn't wind properly.

As the wristwatch is still under guarantee, I am writing to ask for an immediate replacement, and I look forward to receiving this as soon as possible. I am returning the faulty wristwatch under separate cover.

Yours faithfully,

Isabella Santos.
(Ms.)

The information called for in this letter could be extended to include colour, material, size, price and so on, and further templates could be prepared to illustrate different kinds of complaint: perhaps an angry follow-up when no satisfaction has been obtained, and then a cry of anguish after six months' wrangling over the problem. These would draw on the same data as originally supplied by the students, and could be printed one after the other – or possibly spread over more than one session (although this would entail completing the interrogation file from scratch each time).

Other possible applications for mailmerging of letters in this way are: job applications, letters to the school or the doctor, thank you letters, invitations, letters home, sick notes. In addition to letters, this format could be used for guided stories, creative writing and other kinds of written work.

The object of using mailmerge in this way would be to look at register as a means of conveying feelings and emotions, as well as simply examining the standard form of certain kinds of writing. It could also help to give the students confidence by offering a structured approach to letter- or other forms of creative writing.

Pre-computer work:

- discussion of the content of the type of letter which is to be written;
- discussion of the list of questions which will be used in the interrogation file – the discussion should be around the type of answer required, the precise answer being a matter for the students themselves to decide;
- role-play of the situation to be created in the letter (in the example of a letter of complaint, the student would take the wristwatch to the shop personally and ask for a replacement);
- discussion of what might be the content and register of any succeeding letters.

Follow-up exercises:

- show students the templates, so that they can see for themselves the skeletal structure of the letter;
- ask them to create a template of their own on similar lines (perhaps the next stage in a chain of letters on the same theme);
- ask the students to write their own letter on the same theme, either wordprocessed (without a template!) or handwritten;
- get students to create templates for each other (this activity would have to be carefully vetted before the templates were passed on to a second group).

To prepare:

- template(s) and interrogation file(s);
- printed list of questions from the interrogation file(s);
- a worksheet explaining how to load and run the mailmerge files, and how to answer the questions from the interrogation file(s).

NOTES:

Many wordprocessors (including the most popular ones for the BBC) have a mailmerge facility as "standard"; some (like **Wordstar**) have mailmerge as an "optional extra" which has to be bought as a separate package; only the very simplest do not offer a mailmerge facility at all.

The use of mailmerge as described above is rather a specialised one, though it is very easy to use once you understand the principles behind it. More conventional uses for mailmerge are the production of personalised letters, and the printing of address labels (continuous self-adhesive labels can be bought to use with printers which have a "tractor feed" mechanism).

Any group of people to whom you regularly send the same letter (for example, club members, committees, staff) can benefit from mailmerge: all that is needed is a file containing their names and addresses, and any other information which will be used in the letter (membership number, possibly, or a form of address - you might write to Jane Smith as "Jane", whereas Richard Jones is "Mr. Jones"). The data file is usually constructed like this:

Richard Jones, 15 Highbury Mount, Newstead, "London, SE15 3PR", Mr Jones
Jane Smith, The Larches, Normanton Avenue, "Cambridge, CB3 4UZ", Jane
Mr. & Mrs. E. Thompson, 12 Golden Road, London, W12 4QT, Edward and Susan

You will need to check your own wordprocessing handbook for precise details about your own set-up, but the general principles are:

- every complete set of information is called a "record": there are three records in the example above;

- every separate piece of information within a record is called a "field": in the example, there is a name field, followed by three address fields, and one field for the form of address;
- every record must contain the same number of fields, typed in the same order;
- the fields are separated by commas, and every record is ended by pressing the RETURN key;
- if there are commas inside the fields (for example, in "Cambridge, CB3 4UZ"), the whole field must be enclosed in inverted commas (because the program thinks that a comma is the start of a new field!).

The names and addresses must be held in a separate file from the template of the letter you are sending. This means that once you have typed the list of names and addresses, you can use it with any number of different templates, and you will only need to update it from time to time when somebody moves house, or people need to be added to or removed from the list.

You may already have the information about names and addresses stored in a database. In this case, many wordprocessors provide facilities for you to draw the data directly from the database without any need for re-typing. This is probably not worthwhile if the amount of data is small, but if you have, say, a thousand members to write to, it is worth the effort of converting the database information into a data file for your wordprocessor.

TASK 30

Wordprocessor task:
spellcheck/dictionary

Language task:
spelling

Many wordprocessors have a spellcheck facility, which may be an integral part of the wordprocessing program, but is very likely to be an optional extra. The spellcheck program may even be an entirely independent program, produced by a different publisher, so it is worth shopping around to see whether there is such a program *which is compatible with the wordprocessor that you are using*.

If you have not met a spellcheck program before, read the *NOTES* below for more details about how it operates.

This is a free composition exercise, backed up with the spelling-check facility. Before starting on the task, students should be aware of what a spellcheck program can and can not do, and you may like to prepare a demonstration text (or use the one which is supplied with the program) so that they can experiment with the facilities offered before starting on a text of their own.

To prepare:

- a simplified version of the spellcheck handbook, describing all the functions which you want students to use;
- a demonstration text to experiment on.

NOTES:

A spellcheck or dictionary program scans your prepared wordprocessor text and reports back on words which it does not recognise. These words may be:

- *words which would not appear in any dictionary, such as proper names;*

- words which are correctly spelled, but which do not feature in the spellcheck program's dictionary;
- words which are wrongly spelled in the text.

The number of words in the second category will depend on how large the spellcheck dictionary is. A reasonable dictionary would have a minimum of six thousand words, but the larger ones would include some tens of thousands. Even a small dictionary is likely to recognise most of the words language students will want to use.

After checking through the text, the program reports back on each unrecognised word, and waits for instructions on how to handle it. If the word is correctly spelled, there are several options open:

- the word can be added to the main dictionary, so that the program will recognise it when it is found in future texts;
- it can be added to a "personal" dictionary, and when the spellcheck program is run on a future occasion, the user can ask for this personal dictionary to be searched in addition to the main dictionary;
- it can be "ignored" – that is, the word remains unchanged in the text, and the program accepts it as correct; any further occurrences of the word in the same text will also be accepted;

If the word is wrongly spelled, it can be corrected in one of two ways:

- the user can type in the correct version at the keyboard, and the program alters every occurrence of that word in the text;
- the user can ask the program to supply a list of possible alternatives to choose from, and one of the list may be used to replace the wrong spelling in the text: if, for example, the user typed "alterantive", the program might offer the following list from which the correct word could be chosen:

alternate
alternative
alternatively

A spellcheck program is of value to anyone who uses a word-processor extensively. It is very easy to let typing errors and spelling mistakes slip through unnoticed, and a dictionary helps with proofreading. However, there is one case where a spellcheck will not help, and that is where the spelling of a word is correct, but the word has been wrongly used. For example,

The children went out to the park with there mother.

The spellchecker will happily accept "there", because such a word exists. The program has no idea whether what you have typed makes sense: it will merely tell you about the words it does not "know".

Appendix A: CHOOSING A WORDPROCESSOR

In the present economic climate, one of the major factors in choosing a wordprocessor for educational purposes must be cost.

It is generally true that you get what you pay for, but an expensive wordprocessing program may be more complex and offer far more sophistication than you or your students need, and this may make it difficult to use, which in the long run is counter-productive. On the other hand, a very cheap program can be even more difficult to use, offer inadequate facilities, and be less reliable. You may also want to allow for the fact that, once you are familiar with wordprocessing, you (and your students) will probably "grow into" a more sophisticated program. And it could be that, in addition to using the wordprocessor with students, you will want to use it yourself for preparing materials, and perhaps it will be used for administrative purposes as well. It is therefore advisable to look at as wide a range as possible of wordprocessors compatible with your computer system¹, and make the consideration of cost one of the final elements in your choice, rather than one of the first.

WYSIWYG and "preview mode"

Ideally, you should be looking for a WYSIWYG wordprocessor. WYSIWYG (pronounced "wizywig") is not the brand-name of a program, but it represents an important concept in wordprocessing:

What You See Is What You Get

that is, what you see on the screen as you type is what you will see on paper when you print your text. This is not always the case with wordprocessors, especially cheaper ones (although some of the more expensive ones are not much better). There are degrees of WYSIWYG-ness, and you will probably have to compromise a little.

1. The question of compatibility is not considered here. You will need to take factors into consideration such as whether you have a network or standalone machines, and whether you want a disc- or chip-based wordprocessor. Once you have a clear appreciation of the features you will need in your wordprocessing program, you will be in a better position to ask the right questions about the programs which are compatible with your system.

If you use a sophisticated piece of software like *Word* on a Nimbus computer, or *Interword* on the BBC, you will be able to see underlining, bold print, italics and so on actually appearing on the screen as you type, and everything will be in its correct position on the screen. *Edword* is a simple-to-use educational wordprocessor for the BBC which is almost completely WYSIWYG. However, cheaper machines (and cheaper wordprocessors) will probably not be able to cope with such sophistication, and you will have to settle for codes which instruct the printer to print the effects you want to see on paper. It is very difficult to visualise what your printed text will look like when the screen is littered with odd-looking codes, but you should be able at least to preview your text before you print it, and see most of the effects you want on screen at the preview stage. *Wordwise +* is one of the BBC wordprocessors that allows you to do this.

One of the first questions to ask therefore, is whether or not (or to what extent) the wordprocessor is WYSIWYG, and if it is not, whether there is a "preview facility" which will allow you to see how your text will look on paper before you decide to print it. If possible, you should use a number of wordprocessors yourself before buying one. Choose a text which uses several useful formatting features (say, underlining, tabbing and justification), and try out the same text on each wordprocessor you are considering.

Editing

The next consideration is the editing method.

Some wordprocessors have a separate "typing mode" and "editing mode". What this means is that when you are typing your text, you will not be able to correct any mistakes that you see on the screen without going into the "editing mode" first. This is not only inconvenient, but can become extremely irritating when all you need is a quick correction. One example of a wordprocessing program like this is Gemini's *Desk-mate*, which has the added complication of a system of line-numbering which is very difficult and confusing to use.

You should be able to switch from typing to editing text without any difficulty. You should at any time be able to move the cursor to any part of your text in order to make alterations: this is called "screen editing", as opposed to the *Desk-mate* system, which is "line editing".

It might also be useful to find out whether the normal operating mode (the "default mode") of the wordprocessor is "insert" or "overtyping". (If you are not sure of the meaning of these terms, you will find a full description in Task 3). "Insert" is probably the more convenient default mode, simply because most alterations involve adding letters to the original text, or taking away letters

from it; "overtyping" is better suited to those rare alterations where the number of letters is the same before and after the change. If you have a particular interest in the kind of work which involves overtyping (for example, Task 12), avoid *Interword* on the BBC, as this has no overtype mode.

If your computer has "function keys", these will probably be used for some of the editing procedures. Some wordprocessors take this to extremes, however. *View* (BBC Micro) uses the function keys for one set of commands, and the function keys in conjunction with the CONTROL key for a different set of commands, and the function keys in conjunction with the SHIFT key for a third set. There is a "function key strip" (a piece of card printed with reminders of the different commands), but it is still difficult to remember which command is which, and easy to confuse one command with another very similar one.

The alternative to the function key method is the "control code", where editing (and printing) commands are issued by pressing the CONTROL key in combination with one or more alphabetic keys. Although this sounds less satisfactory than function keys, it can be easier to learn as long as the control codes are reasonably mnemonic: if CTRL P means "**P**rint the text" and CTRL H means "**H**elp", there is no problem. Some wordprocessors which use control codes are far from mnemonic, however, *Wordstar* being a prime example. If your wordprocessor is not WYSIWYG, some of the control codes (or function key codes) will appear on the screen along with the text you are typing, and this can look rather strange until you get used to it. For example, the following sentence in which the word "very" is to be underlined (when printed on paper) will appear on a *Wordstar* screen like this:

This is a ^Svery^S strange sentence.

(The ^S symbols represent the control code for switching underlining on and off on *Wordstar*).

Only a really WYSIWYG program will avoid this altogether.

One useful feature to look out for is the on-screen menu. Here *Wordstar* is very helpful, showing a "shopping list" of editing commands on screen so that you do not need to overload your

memory while learning the program, once you are familiar with the commands, the help menu can be withdrawn (there are three levels of help available), leaving more space on the screen for your text.

Another program which uses this approach is *Word* (Nimbus), in which many of the procedures can be carried out either through the network of menus or by a system of control codes, or with a mouse (an add-on piece of hardware which is held in the hand). The *Wordwise* family of wordprocessors, has a dual system where the editing is carried out by control codes, but printing, previewing and other commands are achieved from a menu which is easily obtained with a single key-press.

**40 columns,
80 columns ...
or large print?**

Most "professional standard" wordprocessors do not offer a choice of on-screen print size – they are 80-column only. This means that the length of a line on the screen can be the same as the length of line which you will normally expect to use on A4-width paper, with a reasonable margin at each side. You can, of course, have longer lines or shorter ones, but the standard page is A4-width. The advantage of this is that it is more nearly WYSIWYG than other screen formats. The possible disadvantage from a literacy point of view is that the on-screen print-size is quite small.

If you are thinking in terms of an 80-column wordprocessor, be sure that the monitor you have will cope adequately with the small characters. You will need at least a *good* medium resolution monitor (some of the cheaper "medium resolution" monitors are little better than television sets – Microvitec is a good make) for this; low resolution monitors are unsatisfactory, as are ordinary televisions, because the small print looks fuzzy and is very tiring on the eyes. It is much better to have a good monochrome (black-and-white or green screen) monitor than a poor colour one.

Some wordprocessors offer a choice of 40- or 80-column screen formats. *View* and *Interword* are in this category. The 40-column screen uses a print-size similar to the one commonly used in computer games. This is much easier to read, especially for inexperienced readers, but the disadvantage is that a full line on the screen only occupies half the width of an A4 page, and this can cause problems when viewing the text (in the case of *Interword*) or in printing it (in the case of *View*).

The *Wordwise* family provides only one mode for text entry, and that is 40 columns, although the natural paper-print format is 80 columns. This makes for a rather odd situation where two

complete lines of text on screen will be printed as a single line on paper. It is possible to preview the text as it will appear on paper by using the preview option on the menu, but it can sometimes be very confusing while you are actually typing the text, especially when using unusual page layouts, like tables or charts.

Some programs provide a large-text option in addition to the 40-column mode. *Mini-Office* is one of the cheaper wordprocessors, and (in addition to a 40-column screen option) it allows you to type in 20-column mode, giving very large print on the screen. For beginner readers, this size of print can be an advantage, but what you see on the screen bears no relation to what you will get on paper, and it is therefore difficult to accomplish anything other than very simple "straight" typing.

The large-print facility of *Mini-Office* uses the "expanded" typeface which is easily obtained on a BBC machine; this is very much like the large print which you see on a teletext television (Ceefax or Oracle), and, despite its size, it is not particularly easy to read because of the angular shape of the letters. Another program which offers a large-print option is *Folio* (again for the BBC), but here the typeface has been carefully defined with early readers in mind, and is very easy to read. An additional bonus is that *Folio* is almost completely WYSIWYG.

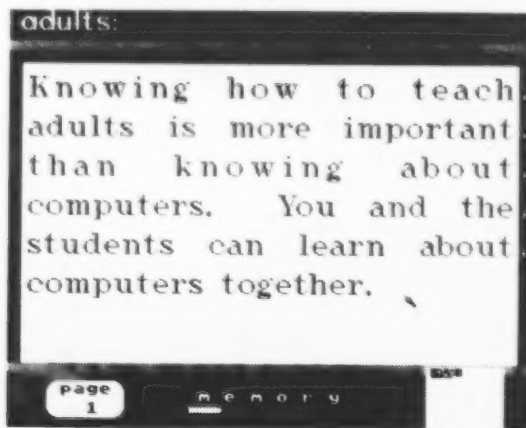


Fig. 10. An excellent wordprocessor for beginner readers and writers. *Folio* (BBC) in "news" mode.

The large print options, although they can be an advantage in the very early stages of reading and writing, are not really suitable for the exercises in this book, which were intended for use with at least a 40-column screen. *Folio*, however, while it does not have many of the advanced wordprocessor facilities which the exercises in this handbook would require, can still be highly recommended for general use with literacy students.

Printing

Wordprocessors handle printing in a variety of different ways. The simplest will allow you only to print a complete text. A more sophisticated one will let you choose to print individual pages of text rather than a whole document. Others will also print any selection of text from a document, which is good if you just need a single paragraph or the odd sentence, but have no need to print the whole document.

If you are going to use your wordprocessor for administrative work, it would probably be useful to have a program that allowed you to do "background printing" – that is, you can continue to type or edit text while a document is being printed. This can be very handy if you need to print a long document, because you can be using the wordprocessor for another purpose while the printing is going on.

Most wordprocessors will allow you to print in a large typeface (provided that your printer has this facility), but again "expanded print" is not very legible. *Folio*, on the other hand, will print on paper in the same beautifully-defined character-set that you see on screen. *Folio* also allows you to choose between three excellent typefaces (including one which looks like newsprint), with three print-sizes and even a facility which allows you to print the smaller typefaces in two or three columns across an A4 sheet.

Other wordprocessor features

Some of the other features of wordprocessing programs are discussed in some detail as they occur at various points throughout the book. In particular, you may wish to look at the following tasks for additional information before making a final choice of wordprocessor:

Tasks 13/14	tabs
Task 19	block move
Task 21	search and replace
Task 25	block copy/windows
Task 27	glossary
Task 29	mailmerge
Task 30	spellcheck

Desktop publishing

Many groups of adult students would enjoy producing a newsletter or magazine, and a wordprocessor is a good start in this direction.

Using a wordprocessor, text can be written, formatted and printed in the desired size and shape, then individual articles can be put into position on the page using traditional "cut and paste" techniques. The wordprocessor gives a slightly more professional appearance than using a typewriter.

The next stage, however, is to use a desktop publishing program which allows its users to:

- produce articles on a wordprocessor, choosing from a selection of typefaces similar to those available to newspaper editors
- create illustrations using a "painting" program
- design a page layout
- transfer the finished articles and pictures into appropriate positions on the page
- print out the pages on a dot matrix printer.

It is usually possible to write articles using an ordinary wordprocessor, then incorporate these into the page layout at a later stage, which means that many people could contribute to the production over a period of time, using a wordprocessor they are familiar with, and the final version could be the work of a group with particular interest in using the publishing program. If you wish to use a wordprocessor with a desktop publishing program in this way, it would be advisable to check that the programs you wish to use are compatible.

There are now a number of programs available for this purpose, including *Fleet Street Editor* and *AMX Pagemaker* (for the BBC micro), and *NewSPApaper* and *Aldus Pagemaker* (for the Nimbus). The *Aldus Pagemaker* is a professional product (and this is reflected in the price); the others are aimed at the educational and amateur market. The programs vary not only in price, but in flexibility and ease of use as well. Some use a small handheld device called a "mouse" to give commands, and these tend to be easier to use than programs which use only the keyboard – although this adds to the expense, as you need to buy the mouse as well as the program!

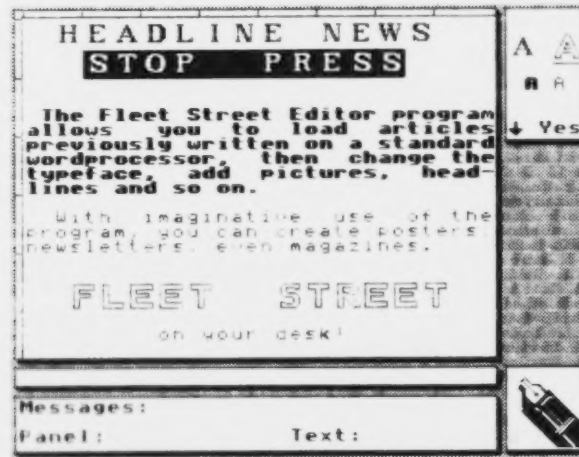


Fig. 11. Desk-top publishing programs offer a wide range of text-handling facilities – and graphics, too. *Fleet Street Editor* (BBC).

Perhaps surprisingly, the degree of skill necessary on the wordprocessor is small – the interest generated by the task in hand is sufficient impetus to get inexperienced people using the wordprocessor without even thinking about it. Students (and teachers, too) are often amazed at their own achievements when working under pressure. Some practice is necessary with the publishing program, however, as this demands a greater level of expertise. It would be advisable to get one or two people thoroughly familiar with the possibilities of the program before embarking on a fullscale project. It is worth bearing in mind that all kinds of skills are needed to produce even a small newsletter, and not all these skills depend on the computer. It is therefore possible for quite a large group of people to be involved in the production, using only one or two computers.

The use of such a program encourages cooperation and enables a group to contribute a variety of skills to the final product. The activity could extend over a period of several weeks, investigating the various stages of the publishing process as well as creating articles, illustrations and advertisements. But possibly the most exciting aspect of desktop publishing is the unique opportunity it provides for teachers and students to learn together.

Appendix B: CHOOSING A PRINTER

The three types of printer most commonly used are the dot matrix, the daisy wheel and the laser printer. The following paragraphs describe briefly the main features of these three types, but there are such wide variations in print quality from printer to printer that you should see a sample printout from the machine you select before purchasing.

Dot matrix printers

This type of printer uses a series of pins which strike the paper in different combinations to produce characters. The quality of the print depends to a large extent on the number of pins used – the smaller the number of pins, the more “computerised” and crude the printout will look. Some of the better dot matrix printers (like the Epson LQ series) produce print which is very close to the quality of a daisywheel.

The Epson FX-85 dot matrix printer
has a fast “draft” mode

Advantages include:

- **flexibility** – dot matrix printers nowadays can provide expanded and condensed print, bold and italic effects, underlining, tiny subscripts and superscripts, and a variety of different international character-sets; it is also possible to print graphics (pictures);
- **price** – a dot matrix is the most economical buy, but it is generally true that you get what you pay for;
- **speed** – this kind of printer is comparatively fast (for example, 180 characters per second for “draft” printing, about 60 characters per second for “quality” printing); this is much faster than a daisywheel, but not as fast as a laser;

The Epson FX-85 dot matrix printer
also has a slower NLQ mode

- **noise** – although still fairly noisy, the dot matrix is still much quieter than a daisywheel printer;

- **compatibility** – Epson have set a standard for dot matrix printers, and most programs which use printers can work with an Epson, so if you buy an Epson (or Epson-compatible) printer, you can avoid many common operating problems.

**Disadvantages
include:**

- **print quality** – which can be poor if you buy a cheap model; the quality will vary from printer to printer, but a printer with an "NLQ" mode should give "Near Letter Quality" results (ie. almost as good as an electric typewriter);

The Epson LQ800 is an exceptionally
good NLQ dot matrix printer

- **speed** – NLQ printing is much slower than draft (but still faster than a daisy wheel printer)

**Daisywheel
printers**

A daisywheel printer prints characters on the paper by means of a metal or plastic wheel with a character at the end of each "spoke". Like a typewriter, the daisywheel prints a complete character when a spoke strikes the paper.

**Advantages
include:**

- **print quality** – the output from a daisywheel printer is the same as that from a good electronic typewriter;
- **flexibility** – different daisywheels can be used to obtain a wide range of typefaces – but the wheel has to be changed each time a new typeface is used;

This is the output from a
C.ITHO F10 daisywheel printer

**Disadvantages
include:**

- **price** – the daisywheel is relatively expensive – what you pay for is the high quality of the output;
- **noise** – this kind of printer is extremely noisy and unpleasant to work with: some daisywheels need a "hood" to quieten them down so that it is possible to work in the same room while the printer is running, and the hood is of course an extra expense;

- **speed** – because there is a limit to how fast the daisywheel can rotate to select each character for printing, this kind of printer can be extremely slow (especially the cheaper versions) – perhaps only 25 characters per second, compared with about 80-100 characters per second with a dot matrix printer;
- **flexibility** – the wheel can accommodate only a limited number of characters, usually only an upper and lower case alphabet, numbers and other common keyboard symbols; anything "unusual" (such as italics) has to be achieved by changing the daisywheel in mid-print (you can ask the wordprocessor to pause while you change the wheel); there is no possibility of expanded or condensed print, or genuine subscripts and superscripts, and international character-sets again need a special wheel; a daisywheel can not cope with graphics and therefore can not be used to print pictures.
- **durability** – the daisywheels are expensive, and the most commonly used character often snaps off the wheel.

Laser printers

A laser printer is unlike either of the other types in that it prints a complete sheet of paper at one go, like a photocopier. There are now many different makes of laser printer on the market, some of which allow more flexibility than others in the choice of fonts (ie. different sizes and styles of print).

Advantages include:

- **print quality** – the range of fonts and excellence of printing mean that the output from a laser printer equals the quality of professional typesetting; this makes it possible to produce printed materials (say, newsletters) to a professional standard;
- **noise** – the laser printer is not quite silent, but it is certainly unobtrusive;
- **speed** – lasers can print a complete A4 page in a few seconds, compared with a minute or so on a dot matrix, and even longer on a daisywheel;
- **flexibility** – depending on which model you choose, the laser printer can handle a wide variety of typefaces and sizes, and some can handle graphics, too;

**The HP Laserjet Plus laser printer
produces a very high quality output**

Disadvantages include:

- **price** – lasers are very expensive, starting at around two thousand pounds, but prices are gradually coming down; however, when considering the price, take into account the fact that, if you learn to use all its features, the laser printer could handle most of the printing needs of a small organisation, and could thus pay for itself within a short time.

Paper

The laser printer uses cut sheets of paper, and can be loaded with headed paper if required. Like a photocopier, it picks up sheets as necessary from a bulk supply placed in a tray mounted on the base of the machine.

The dot matrix and daisywheel printers can also print on cut sheets, using a "friction feed" mechanism like a typewriter. Note however that the use of cut sheets means that somebody has to be on hand to put in a new sheet each time the printer finishes a page. This might not matter for a three-page document, but it quickly gets tiring if you have to print a 100-page report or letters to fifty people on a mailing-list. If you are sufficiently wealthy, you can buy a "cut-sheet feeder" which is an optional extra, and can be loaded with, for example, headed paper in cut sheets.

Most dot matrix and daisywheel printers can also print on "continuous stationery", using a "tractor feed" which is either built into the printer, or can be bought as an optional extra. A tractor feed pulls the paper through the machine via sprockets, which match sprocket-holes along the sides of the paper. This means that you can buy paper in bulk, and tear off individual sheets as they come off the printer. The sheets are perforated top and bottom, and also at the sides, where they are joined to the guides which contain the sprocket-holes.

Ordinary continuous paper (suitable for rough work) has large perforations which make it unsuitable for, say, correspondence, but if you buy "micro-perforated" paper, the perforations are so tiny that, once separated from the guides, each individual sheet of paper looks exactly like a cut sheet. It is also possible to have a letter-head printed on micro-perforated paper (but this might be more expensive in the long run than buying a cut sheet feeder).

Most continuous paper measures 11 inches (280 mms) by 8.5 inches (215 mms), compared with an A4 sheet which is 11.7 inches (297 mms) by 8.3 inches (210 mms). A4 continuous stationery is obtainable, but it is more expensive than the "normal" size.

Appendix C: GLOSSARY OF TERMS

The task numbers refer to the tasks where the relevant features are principally discussed.

- Background Printing** allows you to **edit** one document while printing another
- Block Copy** enables blocks of text to be repeated at any point in a document (*Task 25*)
- Block Delete** enables blocks of text to be removed from a document (*Task 25*)
- Block Move** enables blocks of text to be moved from one point to another within a document (*Task 19*)
- Caps Lock** is a key which turns the upper case alphabet on or off; unlike a typewriter, the CAPS LOCK key on a computer does not usually affect non-alphabetic keys
- Control** is a key which is used in combination with other keys to give commands to the computer (*see also escape*)
- the Cursor** is a block or line (often flashing) which marks the position on the screen where your typing will appear
- the Cursor Keys** are four keys (usually marked with arrows) which allow the **cursor** to be moved through the text on the screen
- a Daisywheel Printer** uses a wheel with characters round its perimeter, and prints by rotating the wheel and striking the appropriate part of the rim against the paper (*see Appendix B for more information about printers*)
- a Database** is a program which allows data (information) to be saved, retrieved and sorted; it is rather like a sophisticated combination of a card-index and a calculator (*Task 17*)
- a Data File** is a wordprocessor **file** which contains information (eg. a list of names and addresses) to be used with a **mailmerged** text (*Task 29*)

Default Mode	is the way your wordprocessor works as soon as you switch it on, until you give it a command to behave differently – for example, some wordprocessors have a justified right margin as "default", whereas others have a ragged right margin (<i>Task 3</i>)
Delete	is the key which erases the letter immediately to the left of the cursor ¹
a Dictionary	(also called spellcheck) is a program which can be used in conjunction with a wordprocessor to assist with the identification of spelling and typing errors (<i>Task 30</i>)
a Dot Matrix Printer	prints characters by striking the paper with a series of pins (<i>see Appendix B for more information about printers</i>)
to Edit	is to alter a wordprocessor text, either to make changes or to add new material
Embedded Commands	are commands which are inserted into a wordprocessor text to affect the way the text will appear when printed (eg. to obtain underlining or bold print)
the Enter key	is the key used to end a paragraph or to create blank lines when the computer is used as a wordprocessor; when the computer is not being used as a wordprocessor, the ENTER key is pressed to end a command to the computer; on some computers, this key is known as the return key
Escape	is a key often used in combination with other keys to give commands to the word processor (<i>see also control</i>)
a Field	is one piece of information contained in a record (<i>see record</i>) – for example, the surname would be a field in a record which contained names and addresses; this expression is used in connection with databases and mailmerge (<i>Task 29</i>)
a File	is a wordprocessor text which has been saved on a disc or tape
to Format a disc	is to prepare it for use; blank discs need to be formatted before they can be used to store files
to Format a text	is to set or alter the layout of the text as it will appear on paper; this can include altering the line-length, the margins, the tab settings etc.
Function Keys	are special keys which can carry out complex functions with one key-press; if your computer has function keys, they will probably be used in wordprocessing
Global Replace	is an option of search and replace which instructs the program to replace automatically all examples of the text you specify (<i>Task 21</i>)

1. Not to be confused with the key marked **DEL** on the Nimbus keyboard, which deletes the character which the cursor is sitting on!

- the Glossary** is a facility of more sophisticated wordprocessors which allows pieces of frequently-used text to be saved on disc for immediate recall and inclusion into any wordprocessor text (*Task 27*)
- a Hanging Indent** is a form of paragraph (like this one) in which the first line remains flush with the margin while subsequent lines are indented; particularly useful for numbered paragraphs (*Task 17*)
- Insert Mode** is the wordprocessor mode in which new text may be inserted into an existing document, the following text moving over to make room for it (*see also otype*) (*Task 3*)
- Justification** is the format in which extra spaces are inserted into lines of text so that the righthand margin forms a straight vertical line (*see also ragged right*)
- a Laser Printer** uses the same principle as a photocopier to achieve high-quality print (*see Appendix B for more information about printers*)
- to Load a file** is to transfer the contents of a file from a disc into the computer's memory (*see also save*)
- Mailmerge** is a function which allows information from **data files** to be incorporated into documents – for example, for the purpose of writing “personalised” letters to people on a mailing list (*Task 29*)
- to Merge files** is to bring two wordprocessor files together in the same document (*Task 26*)
- a Mouse** is a small handheld box which is connected to the computer and is used to move a pointer over the screen; a mouse can make entering wordprocessor commands very simple
- Overtime Mode** is the wordprocessing mode in which new text typed into the middle of existing text causes the existing text to be overwritten (*see also insert mode*) (*Task 12*)
- Preview Mode** is provided on non-**WYSIWYG** (*see WYSIWYG*) wordprocessors to show on screen what the text will look like on paper
- a Printout** is a paper copy of a wordprocessor file, produced on a **printer**
- a Printer** is a piece of equipment which can be attached to a wordprocessor so that paper copies of wordprocessor texts can be made
- Ragged Right** is a paragraph format in which the right hand margin is “ragged” in the same way as it would be if typed on an ordinary typewriter (*see also justification*)

- a Record is one complete set of information in a **data file** or **database** – for example, in a data file of personal information, each record might consist of a name, an address and a telephone number (*see also field*) (Task 29)
- to Reformat is to rearrange a text in a different way – for example, justifying the paragraphs or altering the line-spacing
- Return is the key used to end a paragraph or to create blank lines when the computer is used as a wordprocessor; when the computer is not being used as a wordprocessor, the RETURN key is pressed to end a command to the computer; on some computers, this key is known as the **ENTER** key
- to Save a file is to store it on disc or tape so that it will be preserved when the computer is switched off (*see also load*)
- Search is a wordprocessor function which identifies every example of a specified piece of text within a wordprocessor file (Task 24)
- Search and Replace is a wordprocessor function which identifies every example of a specified piece of text within a wordprocessor file, and replaces it with another specified piece of text (Task 21)
- a Segment (in **Wordwise** +) is a wordprocessor file which can be in the computer's memory at the same time as the main wordprocessor file, though they may not both be on screen at the same time; it is possible to switch between the main file and any of nine segments at any time (*see also windowing*) (Task 25)
- Selective Replace is an option of **search and replace** which instructs the program to replace selected examples of the text you specify, checking with the user before making the replacement (Task 21)
- Shift is a key which causes alphabetic characters to be typed in upper case so long as the key is held down; it usually operates the same way as SHIFT on an ordinary typewriter
- Spellcheck is a program which can be used in conjunction with a wordprocessor to assist with the identification of spelling and typing errors (Task 30)
- the Tab key is the key used to make the cursor jump to the next **tab-stop** (Task 13)
- Tabbing is a function which assists in the layout of columns of text (Task 13)

- a **Tab-Stop** is a point along the width of the text where the cursor will stop when the **tab** key is pressed (*Task 13*)
- Windowing** allows several different texts to be on screen at the same time (*see also segment*) (*Task 25*)
- a **Wordprocessor** is a machine dedicated to wordprocessing, although the word is often used to mean a **wordprocessing program** as used on a computer
- a **Wordprocessing Program** is a program which temporarily converts an ordinary computer into a **word-processor**
- Word-Wrap** is the facility which causes the wordprocessor to start a new line of text when the current line is full
- WYSIWYG** "what you see is what you get" – what appears on the screen when you type is what appears when you print out a paper copy.

Further copies available from:
Adult Literacy & Basic Skills Unit
Kingsbourne House, 229/231 High Holborn,
London WC1V 7DA
Telephone: 01-405 4017

Price: £3.00

ALBSU
Adult Literacy & Basic Skills Unit